Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

# Nondeterminism in generative systems

Master Thesis

2021

bc. Ján Rosina

Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

# Nondeterminism in generative systems

Master Thesis

| | |
|---|---|
| Study Programme: | Computer Science |
| Field of Study: | Computer Science |
| Department: | Department of Computer Science |
| Supervisor: | prof. RNDr. Branislav Rovan, Phd. |

Bratislava, 2021

bc. Ján Rosina

Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

## THESIS ASSIGNMENT

| | |
|---|---|
| **Name and Surname:** | Bc. Ján Rosina |
| **Study programme:** | Computer Science (Single degree study, master II. deg., full time form) |
| **Field of Study:** | Computer Science |
| **Type of Thesis:** | Diploma Thesis |
| **Language of Thesis:** | English |
| **Secondary language:** | Slovak |

**Title:** Nondeterminism in generative systems

**Annotation:** Deterministic and nondeterministic generative systems are studied in the thesis emphasizing the measures of nondeterminism in derivations of generative systems. Furthermore, the generative power of deterministic generative systems with endmarkers is investigated.

**Aim:** The goal of the thesis is to explore deterministic and nondeterministic generative systems with the emphasis on defining and studying the measures of nondeterminism in derivations of generative systems. An additional goal is to explore properties of families of languages generated by deterministic generative systems, in particular the relation of the family of languages generated by deterministic generative systems with endmarker to the family of recursively enumerable languages.

| | |
|---|---|
| **Supervisor:** | prof. RNDr. Branislav Rovan, PhD. |
| **Department:** | FMFI.KI - Department of Computer Science |
| **Head of department:** | prof. RNDr. Martin Škoviera, PhD. |
| **Assigned:** | 01.09.2020 |
| **Approved:** | 08.02.2021 |

prof. RNDr. Rastislav Kráľovič, PhD.
Guarantor of Study Programme

.......................................... Student

.......................................... Supervisor

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

89441077

## ZADANIE ZÁVEREČNEJ PRÁCE

**Meno a priezvisko študenta:** Bc. Ján Rosina
**Študijný program:** informatika (Jednoodborové štúdium, magisterský II. st., denná forma)
**Študijný odbor:** informatika
**Typ záverečnej práce:** diplomová
**Jazyk záverečnej práce:** anglický
**Sekundárny jazyk:** slovenský

**Názov:** Nondeterminism in generative systems
*Nedeterminizmus v generatívnych systémoch*

**Anotácia:** V práci sú skúmané deterministické a nedeterministické generatívne systémy s dôrazom na miery nedeterminizmu v odvodeniach v generatívnych systémoch. V práci je skúmaná aj generatívna sila deterministických generatívnych systémov s endmarkerom.

**Cieľ:** Cieľom práce je skúmať deterministické a nedeterministické generatívne systémy s dôrazom na definovanie a skúmanie mier nedeterminizmu v odvodeniach generatívnych systémov. Ďalším cieľom práce je preskúmať vlastnosti tried generovaných deterministickými g-systémamí, najmä vzťah deterministických g-systémov s endmarkrom a triedy rekurzívne vyčísliteľných jazykov.

**Vedúci:** prof. RNDr. Branislav Rovan, PhD.
**Katedra:** FMFI.KI - Katedra informatiky
**Vedúci katedry:** prof. RNDr. Martin Škoviera, PhD.

**Spôsob sprístupnenia elektronickej verzie práce:**
bez obmedzenia

**Dátum zadania:** 01.09.2020

**Dátum schválenia:** 08.02.2021

prof. RNDr. Rastislav Kráľovič, PhD.
garant študijného programu

..................................... študent

..................................... vedúci práce

# Abstrakt

Táto práca sleduje dva hlavné ciele pri skúmaní generatívnych systémov. Prvým je skúmať silu deterministických generatívnych systémov s endmarkerom, kde sme dokázali rovnosť s triedou rekurzívne vyčísliteľných jazykov. Druhým je definovať a skúmať výpočtové miery nedeterminizmu v generatívnych systémoch. V práci uvádzame dve takéto miery. Prvá meria počet nedeterministických rozhodnutí vzhľadom na dĺžku odvodenia. Pri tejto miere sme ukázali, že pre ľubovoľný rekurzívne vyčísliteľný jazyk existuje generatívny systém s ľubovoľne pomaly rastúcim horným ohraničením zložitosti. Druhá definovaná miera uvažuje počet nedeterministických rozhodnutí pri odvodzovaní slov danej dĺžky. Vo všeobecnom prípade, teda pre rekurzívne vyčísliteľné jazyky, dostávame lineárny horný odhad vzhľadom na dĺžku slov. Pre unárne jazyky a jazyk $\Sigma^*$ dostávame logaritmickú hornú hranicu a pre rekurzívne jazyky horné ohraničenie súvisí s počtom slov danej dĺžky patriacich do daného jazyka.



**Kľúčové slová:** generatívne systémy, determinizmus, nedeterminizmus, generatívne systémy s endmarkerom

# Abstract

This thesis has two main goals in the research of generative systems. The first one is to investigate the power of deterministic generative systems with endmarker for which we prove equality to the family of recursively enumerable languages. The second one is to define and study computational measures of nondeterminism in generative systems. We introduce two such measures. First of them measures the number of nondeterministic decisions in relation to the length of the derivation. For this measure we show that for an arbitrary recursively enumerable language there exists an equivalent generative system with arbitrarily slowly increasing upper bound function of complexity. The second measure considers the dependency of the number of nondeterministic decisions on the length of the derived words. In the general case, for recursively enumerable languages, we obtain linear upper bound function with respect to the length of the words. For unary languages and language $\Sigma^*$ we obtain logarthmic upper bound and upper bound function of recursive languages is related to the number of words of a given length belonging to a given language.

**Keywords:** generative systems, determinism, nondeterminism, generative systems with endmarker

# Contents

**1 Introduction** **1**

**2 Definitions and known results** **3**

2.1 1-a-transducers and generative systems . . . . . . . . . . . . . . . . . . . . 3

2.2 Deterministic g-systems and deterministic g-systems with endmarker . . 4

**3 Deterministic g-systems and prefixes** **7**

**4 A generative power of deterministic g-systems with endmarker** **9**

**5 Measuring nondeterminism in g-systems** **37**

5.1 Considering the number of generative steps . . . . . . . . . . . . . . . . . . 37

5.2 Considering the word length . . . . . . . . . . . . . . . . . . . . . . . . . 45

**6 Conclusion** **63**

# List of Figures


4.1 Phase 1 of \$dgs G generating $L \in \mathcal{L}_{RE}$ which is responsible for the derivation of $w_0\$ \Rightarrow_G^* P_2A^mSHw_1SA^m\$$, where $m$ is the number of steps on which TM A accepts word $w_0$ and $w_1$ is the successor of $w_0$ in rlo. . . . . 13

4.2 Phase 2 of \$dgs G generating $L \in \mathcal{L}_{RE}$ in which are derived subwords on which will be TM A simulated later, so the derivation $P_2A^mSHw_1SA^m\$ \Rightarrow_G^* P_3A^mH\overline{w_1}S...S\overline{a_k^m}S\$$ is made. In the yellow part the last terminal subword is copied, in the blue part this word is incremented in rlo, the green part is responsible for comparison of the length of this word to $m$ and the pink part corresponds to the last step of this phase in which terminal subwords are converted to their double track versions. . . . . . . . . . . . . . . . . 14

4.3 Phase 3 of \$dgs G generating $L \in \mathcal{L}_{RE}$: G works on the sentential from $P_3A^mH\overline{w_1}S...\overline{a_k^m}S\$$. The first of $w_1, ..., a_k^m$ that is in $L_m(A)$ is made terminal or $P_4A^m\$$ is derived if there is no such subword. One step of TM A is simulated on all derived subwords in the yellow branch, after the simulation of $m$ steps of A G checks whether some of them ends in the accepting configuration using blue part and in the green part the nonterminal $M$ or $C$ resp. is shifted to the beginning of the sentential form. . . . . . . . . . 15

4.4 Phase 4 of \$dgs G generating $L \in \mathcal{L}_{RE}$: the number of nonterminals $A$ - simulated steps of TM A - is increased and the second counter is rebuilded so $P_4A^m\$ \Rightarrow_G^* P_2A^{m+1}SHSA^{m+1}\$$ is derived. . . . . . . . . . . . . . . . 16

5.1 Part of M' which is responsible for making $\left.2^{2^{\cdot^{\cdot^{\cdot^{2}}}}}\right\} m$ purely deterministic generative steps for m simulated steps of G. . . . . . . . . . . . . . . . . 43

5.2 G-system that can derive any $w \in \Sigma^*, |w| = n$ using $log(n)$ nondeterministic decisions. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 48

5.3 Phase 1: G uses nondeterminism to derive $\sigma \Rightarrow_G^* P_2A^nP_3ASA^nS$ for arbitrary $n$. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 50

5.4 Phase 2: G deterministically derives $P_2A^nP_3ASA^nS \Rightarrow_G^* P_3A^{k^n}SA^nS$ for $k = |\Sigma|$. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 51

5.5 Phase 3: G derives all words from $\Sigma^n$ for a given $n$ and converts them to the double track version so TM A can be simulated on them. More precisely, this phase is responsible for the derivation of $P_3A^{k^n}SA^nS \Rightarrow_G^* P_4C\overline{a_1^n}\$\overline{a_2a_1^{n-1}}\$...\overline{a_k^n}\$$. . . . . . . . . . . . . . . . . . . . . . . . . . . 52
5.6 Phase 4: G uses nondeterminism to derive subword $A^m$ at the beginning of the sentential form for arbitrary $m$. . . . . . . . . . . . . . . . . . . . . . . . 53
5.7 Phase 5: G simulates TM A on every word of length $n$ and then the $m^{th}$ of the accepted words is made terminal, all others are deleted. . . . . . . 53

# Chapter 1

# Introduction

Grammars represent together with automata and closure operations three major approaches to the study of formal languages and families of languages. In order to satisfy the need of having a general theory of grammars similar to abstract families of languages and abstract families of automata, generative systems were introduced [3].

As one might expect, families of recursively enumerable languages and languages generated by generative systems are equal. However, it turns out that by prohibiting nondeterminism we obtain much weaker model and there exist even some regular languages which cannot be generated by a deterministic generative system [1]. In Chapter 3, we prove one property of infinite languages generated by deterministic generative systems related to the prefix words.

The main cause of this decrease in generative power was identified to be the inability to identify the end of the sentential form. This led to the introduction of a modified model of deterministic generative systems, which maintains a special symbol - endmarker - at the end of the sentential form during the whole generative process. This modification increases the generative power as it enables to generate any recursive language [2]. Relationship between the families of recursively enumerable languages and languages generated by deterministic generative systems with endmarker is the subject of the Chapter 4 of this thesis. We prove that those two families are equal.

The difference between the generative power of nondeterministic and deterministic generative systems raises the question "how much" nondeterminism is actually needed. Two descriptional measures of nondeterminism were defined and studied in the past as an attempt to answer this question [2]. However, descriptional measures do not tell us how much or how often is the nondeterminism used during the generative process. We introduce and study two computational measures in Chapter 5 which measure the number of nondeterministic steps in relation to the length of the derivation and to the length of the derived word. In general, we obtain an arbitrarily slowly increasing upper bound function and a linear upper bound function for the two measures respectively. Moreover, using the second measure we obtain logarithmic upper bound for languages $\Sigma^*$ and unary languages. We prove a better upper bound than in the general case for the case of recursive languages which do not contain "too many" words of length $n$, $n \geq 1$.

# Chapter 2

# Definitions and known results

In this chapter we shall introduce some definitions, notation and known results that are relevant to this thesis.

## 2.1 1-a-transducers and generative systems

**Definition 2.1** *A one-input finite state transducer with accepting states (1-a-transducer) is a 6-tuple* $M = (K, \Sigma_1, \Sigma_2, H, q_0, F)$*, where* $K$ *is a finite set of states,* $\Sigma_1, \Sigma_2$ *are finite alphabets (input and output),* $H$ *is a finite subset of* $K \times \Sigma_1 \times \Sigma_2^* \times K$ *(elements of* $H$ *are called arcs),* $q_0$ *is the initial state and* $F \subseteq K$ *is the set of accepting states.*

**Definition 2.2** *A computation of 1-a-transducer* $M = (K, \Sigma_1, \Sigma_2, H, q_0, F)$ *is a word* $h_1...h_n \in H^*$ *such that:*

$$pr_1(h_1) = q_0 \tag{2.1}$$

$$pr_4(h_n) \in F \tag{2.2}$$

$$pr_4(h_i) = pr_1(h_{i+1}),\ 1 \leq i < n \tag{2.3}$$

*where* $pr_j$ *are homomorphisms on* $H^*$ *defined by* $pr_j((x_1, x_2, x_3, x_4)) = x_j$ *for* $j \in \{1, ..., 4\}$*. The set of all computations of* $M$ *is denoted by* $\Pi_M$*.*

**Definition 2.3** *For any language* $L \subseteq \Sigma_1^*$ *an 1-a-transducer mapping is defined as* $M(L) = pr_3(pr_2^{-1}(L) \cap \Pi_M)$*.*

**Definition 2.4** *A generative system (g-system) is a 4-tuple* $G = (N, T, M, \sigma)$ *where* $N, T$ *are finite alphabets of nonterminal and terminal symbols (not necessarily disjoint),* $M$ *is the 1-a-transducer and* $\sigma \in N$ *is the initial nonterminal.*

**Definition 2.5** *A derivation step of g-system $G = (N, T, M, \sigma)$ is a relation $\Rightarrow_G$ defined by $u \Rightarrow_G v \iff v \in M(u)$.*

**Definition 2.6** *A language generated by g-system $G = (N, T, M, \sigma)$ is the language $L(G) = \{w \in T^* \mid \sigma \Rightarrow_G^* w\}$, where $\Rightarrow_G^*$ is the reflexive and transitive closure of the relation $\Rightarrow_G$.*

**Definition 2.7** *A computation of the 1-a-transducer $M$ on the input word $w$ is denoted by $\alpha_M(w)$. A computation of $M$ on $w$ after $n$ generative steps is denoted recursively:*

$$\alpha_M^1(w) = \alpha_M(w)$$
$$\alpha_M^n(w) = \alpha_M^{n-1}(pr_3(\alpha_M(w))$$

Note that sometimes we may write $\alpha(w)$ instead of $\alpha_M(w)$ when the corresponding 1-a-transducer is apparent from the context.

## 2.2 Deterministic g-systems and deterministic g-systems with endmarker

Now we define a deterministic 1-a-transducer as a special type of the general 1-a-transducer which contains at most one arc for each state and symbol from $\Sigma_1$. Then the definition of deterministic g-system (dgs) follows.

**Definition 2.8** *A 1-a-transducer $M = (K, \Sigma_1, \Sigma_2, H, q_0, F)$ is deterministic iff for each $p \in K$ and $a \in \Sigma_1$ holds: $(p, a, w_1, q_1), (p, a, w_2, q_2) \in H \implies w_1 = w_2$ and $q_1 = q_2$.*

**Definition 2.9** *A g-system $G = (N, T, M, \sigma)$ is deterministic iff its 1-a-transducer $M$ is deterministic.*

**Notation 2.10** *The family of languages generated by generative systems (and deterministic g-systems respectively) will be denoted by $\mathcal{L}_G$ ($\mathcal{L}_{DG}$).*

Note that dealing with deterministic generative systems prefixes of the sentential form are important because g-systems "do not know" anything about symbols that have not been read yet during the particular generative step. In addition, once some symbol is written to the output, it cannot be deleted nor modified in that generative step. It is sometimes useful to order the words in a way that the leftmost symbol is the least important one. Such order is provided e.g. by reversal lexicographic order (rlo) defined as follows:

**Definition 2.11** *Let $\Sigma = \{a_1, ..., a_k\}$ be a given alphabet and let $\prec$ be an order on $\Sigma$ such that $\forall i, j : a_i \prec a_j \iff i < j$.*
*Now we shall define a binary relation $R_l$ over the words in $\Sigma^*$:*

$$\forall u, v\ in \Sigma^*, u = u_1...u_r, v = v_1...v_s :$$
$$R_l(u, v)\ if\ r < s$$
$$R_l(u, v)\ if\ r = s\ and\ u_r \prec v_s$$
$$R_l(u, v)\ if\ r = s\ and\ u_r = v_s\ and\ R_l(u_1...u_{r-1}, v_1...v_{s-1})$$

Clearly relation $R_l$ defined above is a total order on $\Sigma^*$ and we shall call it reversal lexicographic order.

**Notation 2.12** *Let $u, v$ be the words. The fact that $u$ is a prefix (proper prefix, not a prefix, not a proper prefix respectively) of $v$ is denoted by $u \preceq v$ ($u \prec v$, $u \npreceq v$, $u \nprec v$ resp.).*

Most of the above definitions can be found in [3] and [1]. Deterministic g-systems and their descriptive power were subjects of previous research [1]. It was shown that despite the fact that $\mathcal{L}_G = \mathcal{L}_{RE}$ there are even regular languages not in $\mathcal{L}_{DG}$. For example no infinite language $L$ such that $\epsilon \in L$ belongs to $\mathcal{L}_{DG}$. The main cause of this decrease in generative power of dgs was identified to be its inability to identify the end of the sentential form. This lead to defining and studying a modified model of dgs [2] - deterministic g-system with endmarker, which maintains the special symbol at the very end of the sentential form during the whole generative process. The formal definition follows:

**Definition 2.13** *A deterministic g-system with endmarker (\$dgs) G is a deterministic g-system with one special symbol $\$ \notin N \cup T$ which is the rightmost symbol of the sentential form after each generative step of G. It cannot be deleted, nor used elsewhere. The language generated by \$dgs G is $L(G) = \{w \in T^* \mid \sigma\$ \Rightarrow_G^* w\$\}$ and the family of languages generated by all \$dgs is denoted by $\mathcal{L}_{\$DG}$.*

It was proven that $\mathcal{L}_{\$DG} \supseteq \mathcal{L}_{REC}$ [2] and \$dgs constructed in this proof contains the part which simulates arbitrary turing machine A on any (derived) word. We pay attention to this construction because we use it later. The simulation of TM A in G is almost straightforward: sentential form represents the configurations of A during the computation, but in order to simulate shifting the head to the left G has to look two symbols ahead (before writting to the output) and uses states as a buffer for that purpose. Moreover, G uses "double track" symbols - simulation of A is performed on the second track and the first track is used to store the input word so it can be reconstructed after reaching the accepting configuration on the second track.

The double track symbols mentioned are nonterminals denoted by $\frac{a}{b}, a, b \in \Gamma_A$, where $\Gamma_A$ is the tape alphabet of A. We also use notation for double track words composed of double track symbols. Note that in such words the beginnings and the ends of the words from the particular tracks may not match. In such cases the words are padded by blank symbols. The only condition is that the resulting word neither starts nor ends by double track blank $\frac{B}{B}$. For example instead of $\frac{abc}{BaB}$ we write $\frac{abc}{a}$. This leads to ambiguity, because notation $\frac{abc}{a}$ may as well represent sentential form $\frac{abc}{aBB}$ or even $\frac{Babc}{aBBB}$. However, it is not really a problem because all of the following constructions work in a way that the sentential forms they derive can be expressed by such double track notation unambiguously (using the maximal words that do not start or end by $\frac{B}{B}$ symbol).

**Lemma 2.14** *Let $A = (K, \Sigma, \Gamma, \delta, q_0, F)$ be an arbitrary Turing machine. There exists deterministic g-system with endmarker $G = (N, T, M, \sigma)$, where $M = (K_M, N \cup T, N \cup T, H_M, q_{M,0}, F_M)$ such that $\forall w : \frac{B}{B}q_0\frac{w}{w}\frac{B}{B}\$ \Rightarrow_G^m \frac{B}{B}q_F\frac{w}{v}\frac{B}{B}\$ \iff (q_0, w) \vdash_A^m (q_F, v)$ for any $m, q_F \in F, v \in \Gamma^*$.*

*Proof.* The proof follows from the proof of Theorem 3.1 in [2]. Note that $(q_{M,0}, \frac{B}{B}, \epsilon, [\frac{B}{B}]) \in H_M$ in the construction from the mentioned proof. Furthermore, sentential form containts nonterminal $\frac{B}{B}$ at the beginning and at the end after all of those $m$ generative steps of G. We can also assume that M moves on \$ symbol always to $q_{M,0}$ and that $q_{M,0} \in F_M$, because in the original construction there is such state for which those assumptions hold and from the definition of \$dgs no symbol follows after \$. We use these facts later in the constructions which will simulate some Turing machine. □

Note that in the cited proof it is also assumed that the simulated Turing machine ends the computation with head reading the very first symbol of the tape. For that reason, in the following text we assume such Turing machines as well. Similarly, by accepting configuration of TM A we mean the configuration $(q_F u)$, where $q_F$ is some accepting state of A so the head is positioned at the beginning of the tape.

**Notation 2.15** *In the following constructions which simulate some TM A we shall often work with subwords of a form $\frac{B}{B}q_0\frac{w}{w}\frac{B}{B}$, where $q_0$ is the initial state of A. For better readability we use notation $\overline{w} = \frac{B}{B}q_0\frac{w}{w}\frac{B}{B}$, in particular, $\overline{\epsilon} = \frac{B}{B}q_0\frac{B}{B}$.*

# Chapter 3

# Deterministic g-systems and prefixes

In the previous research [1] it was shown that there are languages $L_1, L_2$ such that $L_1 \in \mathcal{L}_{CS} - \mathcal{R}$, $L_2 \in \mathcal{L}_{CF} - \mathcal{L}_{CS}$ such that $L_1, L_2 \notin \mathcal{L}_{DG}$. However, both of these examples were infinite languages containing $\epsilon$. The study [1] suggests that there may be even $\epsilon$-free languages with such properties and as the candidates were mentioned $L_1 = \{ww^R | w \in \Sigma^+\}, L_2 = \{ww | w \in \Sigma^+\}$, where $\Sigma = \{a, b\}$, but no proof was shown.

Furthermore, the study suggests that if it is true that language $\Sigma^+ \notin \mathcal{L}_{DG}$ for binary alphabet $\Sigma$ then we can easily see that $\mathcal{L}_{DG}$ is not closed under $h^{-1}$ and "+".

In this chapter, we prove more general theorem about importance of prefixes in the infinite languages generated by deterministic g-systems which implies that none of the mentioned languages belongs to $\mathcal{L}_{DG}$.

First, let us introduce two useful lemmas which are proved in [1]:

**Lemma 3.1** *Let $L \in \mathcal{L}_{DG}$ be any language. Let $G = (N, T, M, \sigma)$ be deterministic g-system such that L(G)=L. Let $w_1, w_2 \in L$ and let $k, l$ be the integers such that $\sigma \Rightarrow_G^k w_1$ and $\sigma \Rightarrow_G^l w_2$. If $w_1 \preceq w_2$ and $l < k$ then $L$ is finite.*

In other words, dgs generating infinite language derives words in order from shorter to longer (if they have prefix character).

**Lemma 3.2** *Let $M$ be a deterministic 1-a-transducer and let $w_1, w_2$ be two words satisfying $w_1 \preceq w_2$. If $M$ is able to make $n$ generative steps on both inputs $w_1, w_2$ then the following statement is satisfied:*

$$\alpha_M^n(w_1) \preceq \alpha_M^n(w_2).$$

The above lemma confirms the intuition that on identical prefixes deterministic 1-a-transducer works identically. This brings us to the idea that there may be a problem to derive two words which are not prefixes to each other from their common prefix in dgs. For example, let us consider language $\{a^n \mid n > 0\} \cup \{a^n b \mid n > 0\}$. How can any dgs derive from the sentential form $a^i$ both $a^{i+1}$ and $a^i b$ for all $i > 0$? This is the main idea of the theorem from this chapter, in which we prove that it is truly impossible.

Now we prove the lemma about cyclic order of generation of words with the same prefix in deterministic g-systems. This lemma is a key to prove the following theorem.

**Lemma 3.3** *Let $G = (N, T, M, \sigma)$ be a deterministic g-system and let $w_0, w_1 \in L(G)$ be words such that $w_0 \preceq w_1$ and $w_0 \Rightarrow_G^k w_1$ for some positive integer $k$. Let us denote $w_i$ the word such that $w_0 \Rightarrow_G^{ki} w_i$ for all $i$ (if $G$ is able to make corresponding number of generative steps on $w_0$). Then for all words $w_x, w_y$ it holds that $x < y$ implies $w_x \preceq w_y$.*

*Proof.* We prove the lemma by complete induction on $i$. Case $i \leq 0$ is trivial and case $i \leq 1$ follows from the assumption $w_0 \preceq w_1$. Let the statement be satisfied for $i \leq n$, we show that it is also satisfied for $i \leq n+1$. It holds that $w_0 \preceq ... \preceq w_{n-1} \preceq w_n$ and $w_{n-1} \Rightarrow_G^k w_n \Rightarrow_G^k w_{n+1}$. Then from Lemma 3.2 we have that $\alpha_M^k(w_{n-1}) \preceq \alpha_M^k(w_n)$ thus $w_n \preceq w_{n+1}$ and from transitivity of $\preceq$ follows that for all $j < n$ statement $w_j \preceq w_{n+1}$ holds. □

**Theorem 1.** *Let $L$ be an infinite language and let $w, w_1, w_2 \in L$ be the words such that $w \prec w_1$, $w \prec w_2$ but $w_1 \npreceq w_2$, $w_2 \npreceq w_1$. Then $L \notin \mathcal{L}_{DG}$.*

*Proof.* Let us assume by contradiction that there exists dgs G such that $L(G) = L$. From infinity of $L$ and from Lemma 3.1 we have that $w$ is derived before $w_1$ and $w_2$ in G, thus there exist integers $k, l$ such that $w \Rightarrow_G^k w_1$ and $w \Rightarrow_G^l w_2$. Furthermore, G is able to make arbitrary number of generative steps on $w$ so let us denote $v$ a word such that $w \Rightarrow_G^{kl} v$. Lemma 3.3 implies that $w_1 \preceq v$ and also $w_2 \preceq v$ but that would mean that either $w_1 \preceq w_2$ or $w_2 \preceq w_1$ and we have a contradiction. □

**Corollary 3.4** *Let $L_1 = \{ww^R | w \in \Sigma^+\}, L_2 = \{ww | w \in \Sigma^+\}, L_3 = \Sigma^+$ where $|\Sigma| \geq 2$. Then $L_1 \notin \mathcal{L}_{DG}, L_2 \notin \mathcal{L}_{DG}, L_3 \notin \mathcal{L}_{DG}$.*

Note that for sentential forms $u, v$ derived in arbitrary \$dgs holds that $v \preceq u$ or $u \preceq v$ if and only if $u = v$ due to the \$ symbol. This is another point of view at the reason why have \$dgs more generative power than dgs.

# Chapter 4

# A generative power of deterministic g-systems with endmarker

In this chapter, we study the generative power of deterministic generative systems with endmarkers. From the existing results on this topic we know that $\mathcal{L}_{\$DG} \supseteq \mathcal{L}_{REC}$ [2]. In the proof of that fact a \$dgs was used which simulated TM A on all words in a sequence. Nonaccepted words were skipped and those accepted were generated by the \$dgs. This idea obviously does not work on the recursively enumerable languages, because TM A may not halt on certain inputs, so the question whether $\mathcal{L}_{RE} \supsetneq \mathcal{L}_{\$DG}$ or $\mathcal{L}_{RE} = \mathcal{L}_{\$DG}$ remained open. As we show in this chapter, those two families of languages are equal.

In order to prove it we construct an equivalent \$dgs G for any TM A over a terminal alphabet $\Sigma = \{a_1, ..., a_k\}$. Now let us discuss how G works. One possible way of avoiding the nonhalting simulations of A is to simulate increasing but always fixed number of steps. From the definition of \$dgs G can use only information contained in the terminal sentential form $w\$$ to continue the generative process properly and decide which word is next to be processed. So the key of the construction is to find a suitable order of words from the generated language in which they will be derived. It turns out that to order the words by the number of computational steps in which they are accepted by TM A (rlo in case of equality) is a successful idea. G can compute this number $m$ from the derived sentential form $w\$$ by simulating the computation of TM A on $w$ (we assume that $w \in L(A)$ from the fact that $w\$$ was derived). Then it simulates $m$ steps of TM A on "many" words simultaneously and knowing $w$ it can determine the next word to be made terminal. More precisely, we assume such TM A that "reads" the whole input word (its head reads the right blank during the computation) and ends with the head reading the very first symbol of the tape (in case that A accepts or rejects). Thus for fixed number $m$ it holds that no word longer than $m$ is accepted by A on $m$ steps. So by "many" words we mean words $w + 1, ..., a_k^m$ ordered in rlo, where $w + 1$ is the next successor of $w$ in rlo. The leftmost word that is accepted on exactly $m$ steps of A[1] is made terminal and the whole cycle repeats. If no word is accepted then G increments $m$ and simulates $m+1$ steps of A on words $\epsilon, ..., a_k^{m+1}$ etc.

Before we go further into the details of the construction, we define the order in which the words will be derived in G:

**Definition 4.1** *Let $A = (K_A, \Gamma_A, \delta_A, q_{A,0}, F_A)$ be a Turing machine, let $w, v \in L(A)$. Let us denote*

$$m_w = min(m|(q_0 w) \vdash_A^m (q_F w'), where\ q_F \in F_A, w' \in \Gamma_A^*)$$

*the minimal number of steps in which A accepts the word w. We define binary operator $\prec_A$ over $L(A)$ as follows:*

$$w \prec_A v \iff (m_w < m_v) \vee (m_w = m_v \wedge w\ precedes\ v\ in\ rlo).$$

In other words, for a given TM A the relation $\prec_A$ orders the words from $L(A)$ by the minimal number of steps in which they are accepted by A, or in rlo in case of equality. We can easily see that $\prec_A$ on $L(A)$ is a total order, furthermore if $L(A) \neq \emptyset$ then there exists the minimal element (word) in the sense of $\prec_A$.

We denote the set of words accepted by the Turing machine A in $m$ steps by $L_m(A)$. Clearly, $L(A) = \bigcup_{\forall m} L_m(A)$.

To facilitate understanding of the construction we introduce the purpose of individual nonterminals that G uses:

$P_1, P_2, P_3, P_4$ - indicate the current phase of G and are placed at the very beggining of the sentential form. There are only two cases when the first symbol of the sentential form is not one of these: the initial nonterminal $\sigma$ and a terminal word.

$A$ - this nonterminal is used to build and maintain the counters at the beginning and at the end of the sentential form: the first represents the number of steps of TM A simulated and the other determines the maximal meaningful length of the word on which TM A is simulated. Both counters contain the same number of nonterminals $A$.

$S$ - is used as a separator between the blocks.

$C, M$ - flags that some subroutine has ended.

$W$ - flag that some subroutine is in progress

$H, I, L, R$ - these nonterminals are used as "heads" - their position in the sentential form determines which symbol is going to be copied, how many times is some branch of M used, they are used to compare the length of two subwords etc.

[1] We assume that all words accepted on less than $m$ steps of A were generated earlier.

The crucial problem is how to derive the next word in the order given by $\prec_A$ from the derived terminal sentential form $w_0\$$. G does this in 4 phases:

Phase 1: $w_0\$ \rightsquigarrow P_2A^mSHw_1SA^m\$$ - in this phase G computes the number of steps $m$ on which TM A accepts $w_0$.
G simulates the computation of TM A on the word $w_0$. We assume that $w_0 \in L(A)$ because G derived the sentential form $w_0\$$ so $w_0 \in L(G)$. In one generative step of G is one nonterminal $A$ added to both counters and one computational step of TM A is simulated using double track symbols and \$dgs from Lemma 2.14. When simulated Turing machine reaches the accepting configuration nonterminal $M$ is written to the output and the initial word $w_0$ is restored and incremented (in the sense of rlo). In the next steps, flag $M$ is shifted to the left and when it reaches the very first nonterminal, prefix $P_1M$ is replaced by $P_2$ so the phase 2 follows.

Phase 2: $P_2A^mSHw_1SA^m\$ \rightsquigarrow P_3A^mH\overline{w_1}S...S\overline{a_k^m}S\$$ - in this phase all words greater than $w_0$ (in the sense of rlo)[2] of the maximal length $m$ are generated. Again, properties of TM A which we have assumed imply that no longer word can be accepted by A on $m$ steps.
G works in a cycle: the last complete terminal subword in the sentential form is copied, then incremented in the sense of rlo. If it is necessary to extend the copied word by one symbol for that purpose, G compares the length of this word to the length of the counter $A^m$ at the end of the sentential form. If the copied word is not longer than $m$ then it becomes the last complete word which is to be copied. Otherwise this phase is going to be terminated - the block of nonterminals $A$ at the end of the sentential form is deleted as well as the recently generated subword and the nonterminal $M$ is generated and shifted to the left in the following steps. In the last step of this phase, when the sentential form starts with $P_2M$, all the generated subwords are converted to their "double track version" in order to use \$dgs from Lemma 2.14 for simulation of A on them.

Phase 3: $P_3A^mH\overline{w_1}S...S\overline{a_k^m}S\$ \rightsquigarrow w_i\$$, where $w_i$ is the minimal word in the sense of $\prec_A$ from $\{w_1, ..., a_k^m\} \cap L_m(A)$. In case that such $w_i$ does not exist $P_4a^m\$$ is derived in this phase instead. In this phase, $m$ computational steps of TM A on subwords $w_1, ..., a_k^m$ are simulated[3]. The first of the words in the sentential form that is accepted after $m$ steps for the first time is made terminal and G moves to the phase 1. If such subword does not exist then $P_4A^m\$$ is derived and phase 4 follows. The subwords which are accepted on less than $m$ steps are skipped during the simulation.

Phase 4: $P_4A^m\$ \rightsquigarrow P_2A^{m+1}SHSA^{m+1}\$$ - the counter (block of nonterminals A) is

[2]We assume that all words that precede $w_0$ in rlo and are accepted by TM A on $m$ steps have been generated already.

[3]We use \$dgs from Lemma 2.14 on subwords $\overline{w}S$ for that purpose.

incremented by one and the sentential form is modified so phase 2 may follow. From the description of the phase 2 we can see that the derived sentential form $P_2A^{m+1}SHSA^{m+1}\$$ then leads to simulation of $m+1$ steps of TM A on words $\epsilon, ..., a_k^{m+1}$. In order to derive the sentential form containing two counters the number of nonterminals $A$ is increased from $m$ to $2m+2$ and then the middle of this block is found[4] and nonterminals $SHS$ are inserted in that place.

**Theorem 2.** $\mathcal{L}_{RE} = \mathcal{L}_{\$DG}$

*Proof.* We construct \$dgs G for an arbitrary language $L \in \mathcal{L}_{RE}$. There exists a Turing machine $A = (K_A, \Gamma_A, \delta_A, q_{A,0}, F_A)$, such that in each accepting computation of A the configuration when the head reads the right blank is reached and if A accepts, the head is moved to the left so it reads the first symbol of the tape and $L(A) = L$. Let $G_A = (N_{G_A}, T_{G_A}, M_{G_A}, \sigma_{G_A})$, where $M_{G_A} = (K_{G_A}, N_{G_A} \cup T_{G_A}, N_{G_A} \cup T_{G_A}, H_{G_A}, q_{G_A,0}, F_{G_A})$ be the \$dgs from the Lemma 2.14 which simulates a computation of A on a given input. We construct $G = (N, T, M, \sigma)$ as follows:

$$
\begin{aligned}
T =& T_{G_A}\\
N =& N_{G_A} \cup N_{new}, N_{new} = \{A, C, H, I, M, L, R, S, W, P_1, P_2, P_3, P_4\},\ N_{G_A} \cap N_{new} = \emptyset\\
M =& (K, N \cup T, N \cup T, H, q_0, F)\ where:\\
K =& K_{G_A} \cup K_1 \cup K_2 \cup K_3 \cup K_4 \cup \{q_0\},\ where:\\
& K_1 = \{q_{P_1,1}, ..., q_{P_1,11}\}\\
& K_2 = \{q_{P_2,1}, ..., q_{P_2,18}\} \cup\\
& \quad \cup\ \{[P_2, x, i] \mid x \in N \cup T, i \in \{1, ..., 4\}\} \cup\\
& \quad \cup\ \{q_{P_2,a} \mid a \in T\}\\
& K_3 = \{q_{P_3,1}, ..., q_{P_3,17}\}\\
& K_4 = \{q_{P_4,1}, ..., q_{P_4,10}\}\\
& and\ K_{G_A} \cap K_i = \emptyset,\ for\ i \in \{1, ..., 4\}\\
& F = K
\end{aligned}
$$

$$
H = H_1 \cup H_2 \cup H_3 \cup H_4 \cup \{(q_0, \sigma, P_3H\frac{B}{B}q_{A,0}\frac{B}{B}S\$, q_0)\}
$$

$$
\begin{aligned}
H_1 =& H_{G_A} - \{(q, \$, v\$, p) | \forall q, p \in K_{G_A}, v \in (N \cup T)^*\} \cup \{\\
& (q_0, a, P_1\frac{B}{B}q_{A,0}\frac{a}{a}, q_{P_1,1}), \forall a \in T\\
& (q_0, \$, P_1\frac{B}{B}q_{A,0}\frac{B}{B}\$, q_0)
\end{aligned}
$$

[4]This is obtained by shifting nonterminals $L$ from the beginning and $R$ from the end of the sentential form towards themselves as we shall show later in the text.

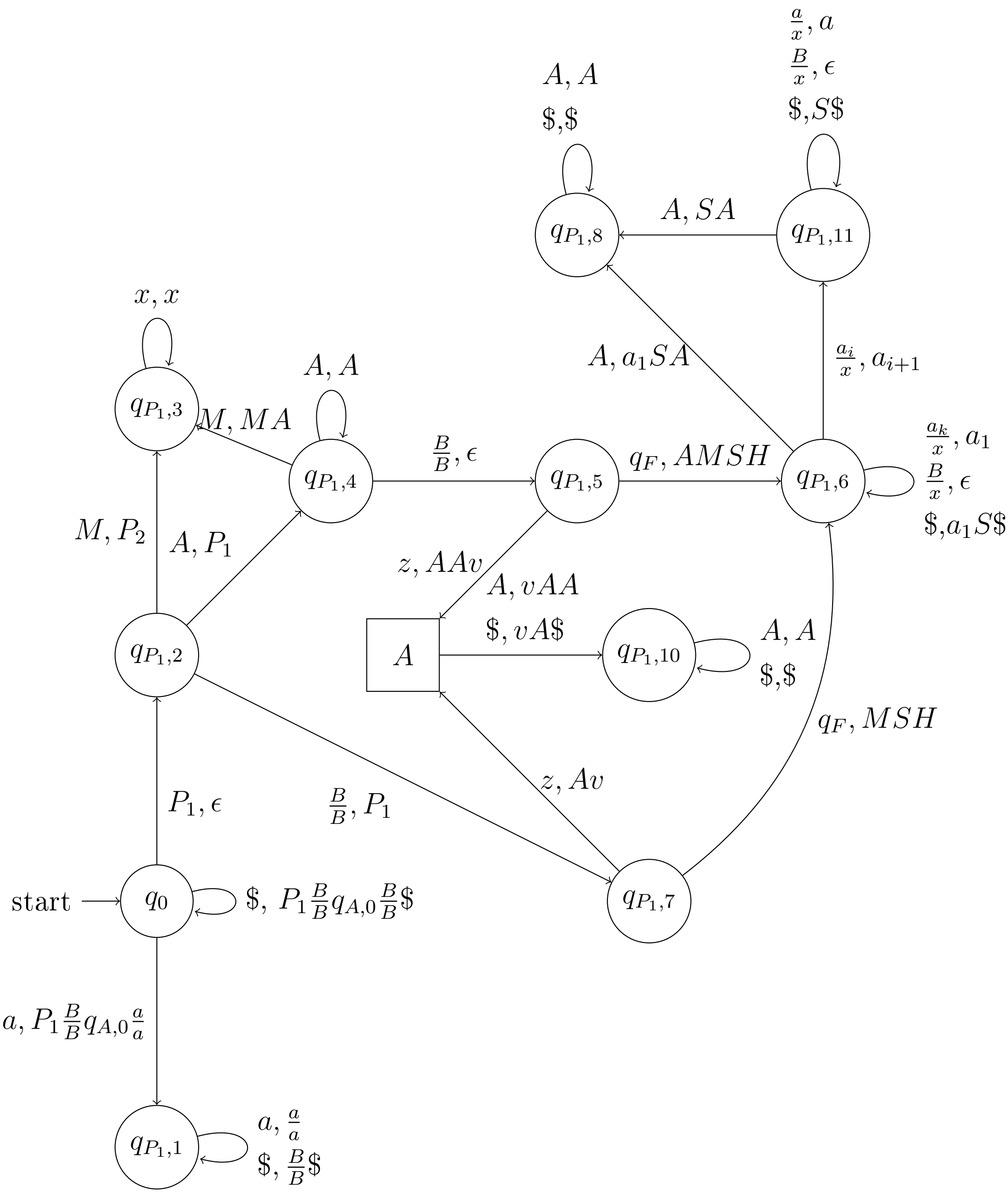


Figure 4.1: Phase 1 of \$dgs G generating $L \in \mathcal{L}_{RE}$ which is responsible for the derivation of $w_0\$ \Rightarrow_G^* P_2A^mSHw_1SA^m\$$, where $m$ is the number of steps on which TM A accepts word $w_0$ and $w_1$ is the successor of $w_0$ in rlo.

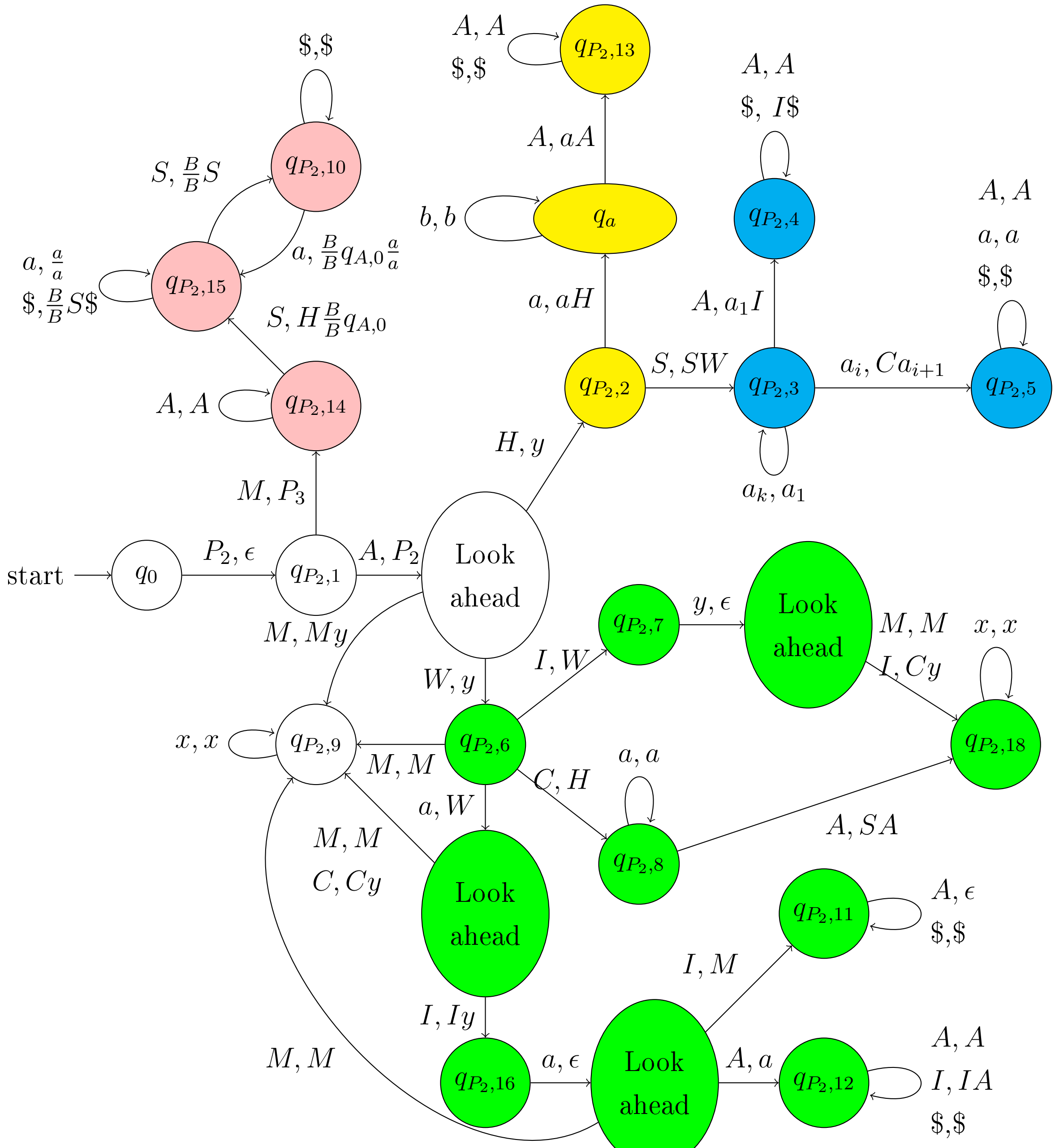


Figure 4.2: Phase 2 of \$dgs G generating $L \in \mathcal{L}_{RE}$ in which are derived subwords on which will be TM A simulated later, so the derivation $P_2A^mSHw_1SA^m\$ \Rightarrow^*_G P_3A^mH\overline{w_1}S...S\overline{a_k^m}S\$$ is made. In the yellow part the last terminal subword is copied, in the blue part this word is incremented in rlo, the green part is responsible for comparison of the length of this word to $m$ and the pink part corresponds to the last step of this phase in which terminal subwords are converted to their double track versions.

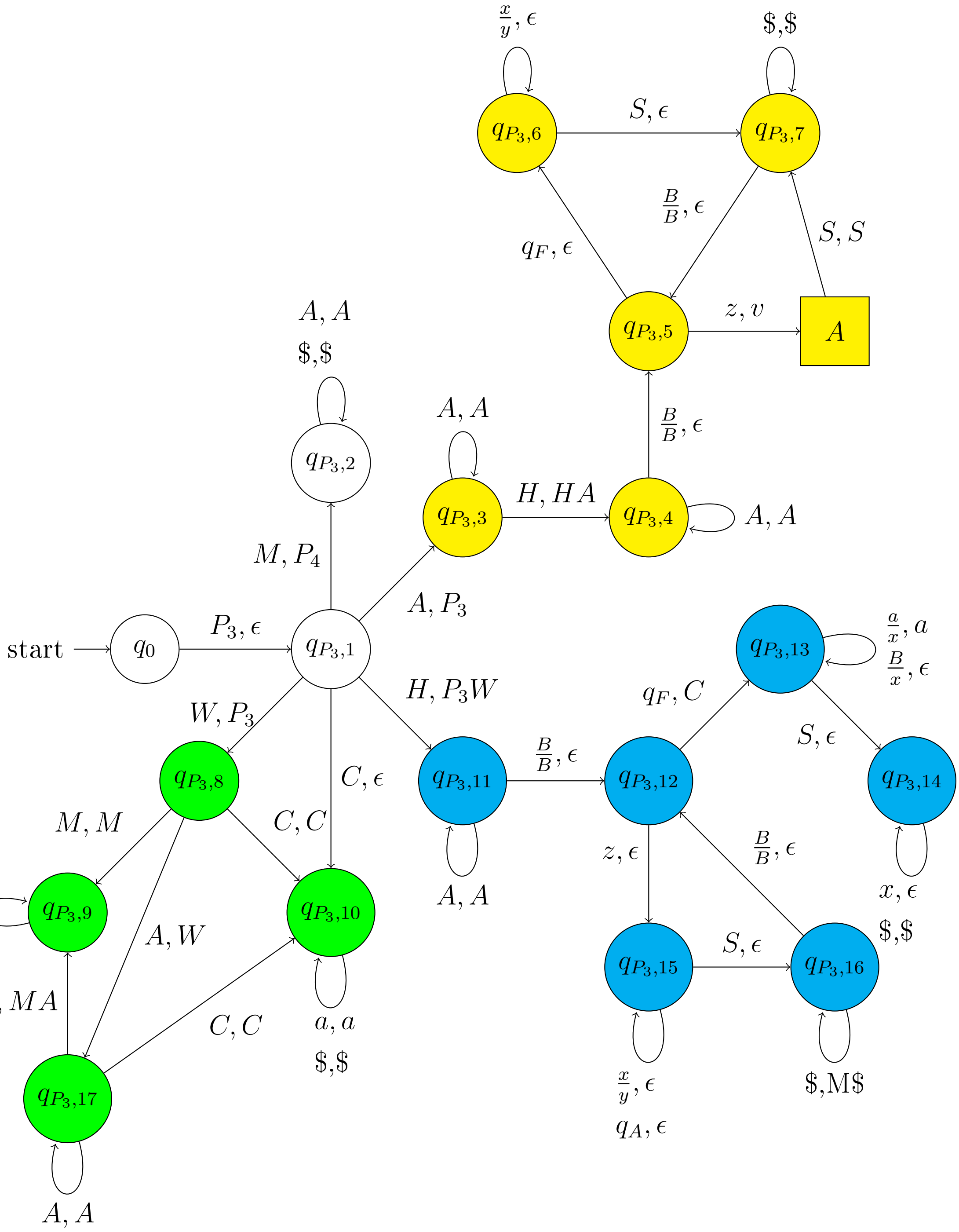


Figure 4.3: Phase 3 of \$dgs G generating $L \in \mathcal{L}_{RE}$: G works on the sentential from $P_3A^mH\overline{w_1}S...\overline{a_k^m}S\$$. The first of $w_1, ..., a_k^m$ that is in $L_m(A)$ is made terminal or $P_4A^m\$$ is derived if there is no such subword. One step of TM A is simulated on all derived subwords in the yellow branch, after the simulation of $m$ steps of A G checks whether some of them ends in the accepting configuration using blue part and in the green part the nonterminal $M$ or $C$ resp. is shifted to the beginning of the sentential form.

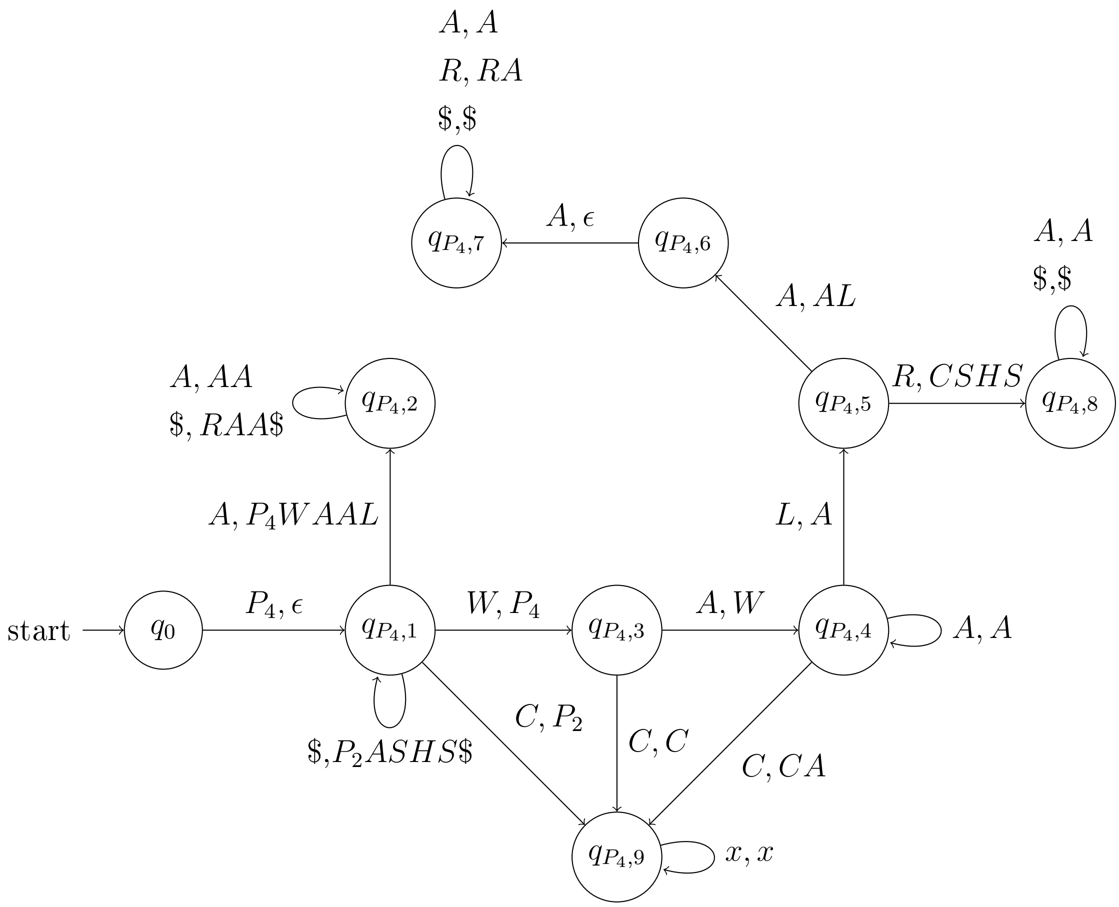


Figure 4.4: Phase 4 of \$dgs G generating $L \in \mathcal{L}_{RE}$: the number of nonterminals $A$ - simulated steps of TM A - is increased and the second counter is rebuilded so $P_4A^m\$ \Rightarrow_G^* P_2A^{m+1}SHSA^{m+1}\$$ is derived.

$(q_{P_1,1}, a, \frac{a}{a}, q_{P_1,1}), \forall a \in T$

$(q_{P_1,1}, \$, \frac{B}{B}\$, q_{P_1,1}),$

$(q_0, P_1, \epsilon, q_{P_1,2}),$

$(q_{P_1,2}, M, P_2, q_{P_1,3}),$

$(q_{P_1,2}, A, P_1, q_{P_1,4}),$

$(q_{P_1,2}, \frac{B}{B}, P_1, q_{P_1,7}),$

$(q_{P_1,3}, x, x, q_{P_1,3}), \forall x \in N \cup T$

$(q_{P_1,4}, A, A, q_{P_1,4}),$

$(q_{P_1,4}, M, MA, q_{P_1,3}),$

$(q_{P_1,4}, \frac{B}{B}, \epsilon, q_{P_1,5}),$

$(q_{P_1,5}, q_F, AMSH, q_{P_1,6}), where\ q_F \in F_A$

$(q_{P_1,6}, \frac{a_k}{x}, a_1, q_{P_1,6}), \forall x \in N \cup T$

$(q_{P_1,6}, \frac{B}{x}, \epsilon, q_{P_1,6}), \forall x \in N \cup T$

$(q_{P_1,6}, A, a_1SA, q_{P_1,8}),$

$(q_{P_1,6}, \$, a_1S\$, q_{P_1,6}),$

$(q_{P_1,8}, A, A, q_{P_1,8}),$

$(q_{P_1,8}, \$, \$, q_{P_1,8}),$

$(q_{P_1,5}, z, AAv, q_{A,z}), \forall z \in N \cup T - F_A;\ where\ (\left[\frac{B}{B}\right], z, v, q_{A,z}) \in H_{G_A}$

$(q_{P_1,7}, z, Av, q_{A,z}), \forall z \in N \cup T - F_A;\ where\ (\left[\frac{B}{B}\right], z, v, q_{A,z}) \in H_{G_A}$

$(q_{P_1,7}, q_F, MSH, q_{P_1,6}), where\ q_F \in F_A$

$(q_A, A, vAA, q_{P_1,10}), \forall q_A \in K_{G_A}, v \in (N_A \cup T_A)^* : \exists p \in K_{G_A} : (q_A, \$, v\$, p) \in H_{G_A},$

$(q_A, \$, vA\$, q_{P_1,10}), \forall q_A \in K_{G_A}, v \in (N_A \cup T_A)^* : \exists p \in K_{G_A} : (q_A, \$, v\$, p) \in H_{G_A},$

$(q_{P_1,10}, A, A, q_{P_1,10}),$

$(q_{P_1,10}, \$, \$, q_{P_1,10}),$

$(q_{P_1,6}, \frac{a_i}{x}, a_{i+1}, q_{P_1,11}), \forall x \in N \cup T$

$(q_{P_1,11}, \frac{a}{x}, a, q_{P_1,11}), \forall a \in T$

$(q_{P_1,11}, \frac{B}{x}, \epsilon, q_{P_1,11}), \forall x \in N \cup T\},$

$(q_{P_1,11}, \$, S\$, q_{P_1,11})$

$(q_{P_1,11}, A, SA, q_{P_1,8})$

$H_2 = \{$

$(q_0, P_2, \epsilon, q_{P_2,1}),$

$(q_{P_2,1}, A, P_2, [P_2, A, 1]),$

$([P_2, y, 1], x, y, [P_2, x, 1]) \ \forall x, y \in (N \cup T) - \{H, W, M\},$

$([P_2, y, 1], H, y, q_{P_2,2}) \ \forall y \in (N \cup T) - \{H, W, M\},$

$(q_{P_2,2}, a, aH, q_{P_2,a}) \ \forall a \in T,$

$(q_{P_2,a}, b, b, q_{P_2,a}) \ \forall a \in T, b \in T \cup \{S\}$

$(q_{P_2,a}, A, aA, q_{P_2,13}) \ \forall a \in T,$

$(q_{P_2,13}, A, A, q_{P_2,13}),$

$(q_{P_2,13}, \$, \$, q_{P_2,13}),$

$(q_{P_2,2}, S, SW, q_{P_2,3}),$

$(q_{P_2,3}, a_k, a_1, q_{P_2,3}),$

$(q_{P_2,3}, A, a_1 I, q_{P_2,4}),$

$(q_{P_2,3}, a_i, Ca_{i+1}, q_{P_2,5}),$

$(q_{P_2,4}, A, A, q_{P_2,4}),$

$(q_{P_2,4}, \$, I\$, q_{P_2,4}),$

$(q_{P_2,5}, A, A, q_{P_2,5}),$

$(q_{P_2,5}, a, a, q_{P_2,5}) \ \forall a \in T,$

$(q_{P_2,5}, \$, \$, q_{P_2,5}),$

$([P_2, y, 1], W, y, q_{P_2,6}) \ \forall y \in (N \cup T) - \{H, W\},$

$(q_{P_2,6}, I, W, q_{P_2,7}),$

$(q_{P_2,7}, y, \epsilon, [P_2, y, 3]) \ \forall y \in T \cup \{A\},$

$(q_{P_2,8}, a, a, q_{P_2,8}) \ \forall a \in T,$

$(q_{P_2,8}, A, SA, q_{P_2,18}),$

$(q_{P_2,18}, x, x, q_{P_2,18}) \ \forall x \in N \cup T,$

$([P_2, y, 3], M, M, q_{P_2,18}) \ \forall y \in T \cup \{A\},$

$([P_2, y, 3], I, Cy, q_{P_2,18}) \ \forall y \in T \cup \{A\},$

$(q_{P_2,6}, a, W, [P_2, a, 2]) \ \forall a \in T,$

$([P_2, y, 2], M, M, q_{P_2,9}) \ \forall y \in (N \cup T) - \{M, C, I\},$

$([P_2, y, 2], C, Cy, q_{P_2,9}) \ \forall y \in (N \cup T) - \{M, C, I\},$

$([P_2, y, 2], I, Iy, q_{P_2,16}) \ \forall y \in (N \cup T) - \{M, C, I\},$

$([P_2, y, 2], x, y, [P_2, x, 2])\ \forall x, y \in T$

$([P_2, y, 4], x, y, [P_2, x, 4])\ \forall x, y \in T,$

$([P_2, y, 4], M, M, q_{P_2,9})\ \forall y \in T,$

$([P_2, y, 4], I, M, q_{P_2,11})\ \forall y \in T,$

$(q_{P_2,11}, A, \epsilon, q_{P_2,11}),$

$(q_{P_2,11}, \$, \$, q_{P_2,11}),$

$([P_2, y, 4], A, y, q_{P_2,12})\ \forall y \in T,$

$(q_{P_2,12}, A, A, q_{P_2,12}),$

$(q_{P_2,12}, I, IA, q_{P_2,12}),$

$(q_{P_2,12}, \$, \$, q_{P_2,12}),$

$(q_{P_2,6}, C, H, q_{P_2,8}),$

$(q_{P_2,6}, M, M, q_{P_2,9}),$

$(q_{P_2,9}, x, x, q_{P_2,9})\ \forall x \in N \cup T,$

$([P_2, y, 1], M, My, q_{P_2,9})\ \forall y \in (N \cup T) - \{H, W, M\},$

$(q_{P_2,1}, M, P_3, q_{P_2,14}),$

$(q_{P_2,14}, A, A, q_{P_2,14}),$

$(q_{P_2,14}, S, H\frac{B}{B}q_{A,0}, q_{P_2,15})$

$(q_{P_2,15}, a, \frac{a}{a}, q_{P_2,15})\ \forall a \in T,$

$(q_{P_2,15}, S, \frac{B}{B}S\frac{B}{B}q_{A,0}, q_{P_2,15}),$

$(q_{P_2,15}, \$, \$, q_{P_2,15}),$

$(q_{P_2,16}, a, \epsilon, [P_2, a, 4])\forall a \in T\},$

$H_3 = H_{G_A} - \{(q, \$, v\$, p) | \forall q, p \in K_{G_A}, v \in (N \cup T)^*\} \cup \{$

$(q_0, P_3, \epsilon, q_{P_3,1}),$

$(q_{P_3,1}, M, P_4, q_{P_3,2}),$

$(q_{P_3,2}, A, A, q_{P_3,2}),$

$(q_{P_3,2}, \$, \$, q_{P_3,2}),$

$(q_{P_3,1}, A, P_3, q_{P_3,3}),$

$(q_{P_3,3}, A, A, q_{P_3,3}),$

$(q_{P_3,3}, H, HA, q_{P_3,4}),$

$(q_{P_3,4}, A, A, q_{P_3,4}),$

$(q_{P_3,4}, \dfrac{B}{B}, \epsilon, q_{P_3,5})$,

$(q_{P_3,5}, q_F, \epsilon, q_{P_3,6})\ \forall q_F \in F_A$,

$(q_{P_3,6}, \dfrac{x}{y}, \epsilon, q_{P_3,6})\ \forall x, y \in \Gamma_A$,

$(q_{P_3,6}, S, \epsilon, q_{P_3,7})$,

$(q_{P_3,7}, \$, \$, q_{P_3,7})$,

$(q_{P_3,7}, \dfrac{B}{B}, \epsilon, q_{P_3,5})$,

$(q_{P_3,5}, z, v, q_A)\ \forall z \in (N \cup T) - F_A,\ \text{where } (\begin{bmatrix} B \\ \hline B \end{bmatrix}, z, v, q_A) \in H_{G_A}$,

$(q_A, S, vS, q_{P_3,7})\ \forall q_A \in K_{G_A}, v \in (N_A \cup T_A)^* : \exists p \in K_{G_A} : (q_A, \$, v\$, p) \in H_{G_A}$,

$(q_{P_3,1}, W, P_3, q_{P_3,8})$,

$(q_{P_3,8}, M, M, q_{P_3,9})$,

$(q_{P_3,9}, A, A, q_{P_3,9})$,

$(q_{P_3,9}, \$, \$, q_{P_3,9})$,

$(q_{P_3,8}, A, W, q_{P_3,17})$,

$(q_{P_3,17}, A, A, q_{P_3,17})$,

$(q_{P_3,17}, M, MA, q_{P_3,9})$,

$(q_{P_3,17}, C, C, q_{P_3,10})$,

$(q_{P_3,8}, C, C, q_{P_3,10})$,

$(q_{P_3,1}, C, \epsilon, q_{P_3,10})$,

$(q_{P_3,10}, a, a, q_{P_3,10})\ \forall a \in T$,

$(q_{P_3,10}, \$, \$, q_{P_3,10})$,

$(q_{P_3,1}, H, P_3W, q_{P_3,11})$,

$(q_{P_3,11}, A, A, q_{P_3,11})$,

$(q_{P_3,11}, \dfrac{B}{B}, \epsilon, q_{P_3,12})$,

$(q_{P_3,12}, z, \epsilon, q_{P_3,15})\ \forall z \in (N \cup T) - F_A$,

$(q_{P_3,15}, \dfrac{x}{y}, \epsilon, q_{P_3,15})\ \forall x, y \in \Gamma_A$,

$(q_{P_3,15}, q_A, \epsilon, q_{P_3,15})\ \forall q_A \in K_A$,

$(q_{P_3,15}, S, \epsilon, q_{P_3,16})$,

$(q_{P_3,16}, \dfrac{B}{B}, \epsilon, q_{P_3,12})$,

$(q_{P_3,12}, q_F, C, q_{P_3,13})$,

$(q_{P_3,13}, \dfrac{a}{x}, a, q_{P_3,13})\ \forall a \in T, \forall x \in \Gamma_A,$

$(q_{P_3,13}, \dfrac{B}{x}, \epsilon, q_{P_3,13})\ \forall x \in \Gamma_A,$

$(q_{P_3,13}, S, \epsilon, q_{P_3,14}),$

$(q_{P_3,14}, x, \epsilon, q_{P_3,14})\ \forall x \in (N \cup T) - \{\$\},$

$(q_{P_3,14}, \$, \$, q_{P_3,14}),$

$(q_{P_3,16}, \$, M\$, q_{P_3,16})\}$

$H_4 = \{$

$(q_0, P_4, \epsilon, q_{P_4,1}),$

$(q_{P_4,1}, A, P_4WAAL, q_{P_4,2}),$

$(q_{P_4,1}, \$, P_2ASHS\$, q_{P_4,1})$

$(q_{P_4,2}, A, AA, q_{P_4,2}),$

$(q_{P_4,2}, \$, RAA\$, q_{P_4,2}),$

$(q_{P_4,1}, W, P_4, q_{P_4,3}),$

$(q_{P_4,3}, A, W, q_{P_4,4}),$

$(q_{P_4,4}, A, A, q_{P_4,4}),$

$(q_{P_4,4}, L, A, q_{P_4,5}),$

$(q_{P_4,5}, A, AL, q_{P_4,6}),$

$(q_{P_4,6}, A, \epsilon, q_{P_4,7}),$

$(q_{P_4,7}, A, A, q_{P_4,7}),$

$(q_{P_4,7}, R, RA, q_{P_4,7}),$

$(q_{P_4,7}, \$, \$, q_{P_4,7}),$

$(q_{P_4,5}, R, CSHS, q_{P_4,8}),$

$(q_{P_4,8}, A, A, q_{P_4,8}),$

$(q_{P_4,8}, \$, \$, q_{P_4,8}),$

$(q_{P_4,4}, C, CA, q_{P_4,9}),$

$(q_{P_4,3}, C, C, q_{P_4,9}),$

$(q_{P_4,1}, C, P_2, q_{P_4,9}),$

$(q_{P_4,9}, x, x, q_{P_4,9}) \forall x \in N,$

Note that $H_1, H_2, H_3, H_4$ correspond to the phases 1,2,3,4 described earlier and are depicted in the Figures 4.1,4.2,4.3,4.4.

The following proof is separated into two parts: first, we show that $L(G) = L(A)$ assuming that G works in particular phases as suggested and then we prove that these asumptions are true by showing the concrete computations of 1-a-transducer M.

Let us assume that G works as follows:

$$w_0\$ \Rightarrow_G^* P_2A^mSHw_1SA^m\$ \qquad (phase\ 1) \qquad (4.1)$$
$$P_2A^mSHw_1SA^m\$ \Rightarrow_G^* P_3A^mH\overline{w_1}S...S\overline{a_k^m}S\$ \qquad (phase\ 2) \qquad (4.2)$$
$$either\ P_3A^mH\overline{w_1}S...S\overline{a_k^m}S\$ \Rightarrow_G^* w_i\$ \qquad (phase\ 3) \qquad (4.3)$$
$$or\ P_3A^mH\overline{w_1}S...S\overline{a_k^m}S\$ \ \Rightarrow_G^* P_4A^m\$ \qquad (phase\ 3) \qquad (4.4)$$
$$P_4A^m\$ \Rightarrow_G^* P_2A^{m+1}SHSA^{m+1}\$ \qquad (phase\ 4) \qquad (4.5)$$

where $w_0 \in L_m(A)$, $w_1$ is the successor of $w_0$ in rlo, notation $\overline{w}$ stands for $\frac{B}{B}q_{A,0}\frac{w}{w}\frac{B}{B}$ and case 4.3 occurs iff $w_i$ is the minimal element (in sense of rlo) of $\{w_1, ..., a_k^m\} \cap L_m(A)$ and case 4.4 occurs iff $\{w_1, ..., a_k^m\} \cap L_m(A) = \emptyset$. In other words, if in the phase 3 at least one of the subwords in the sentential form is accepted by TM A in $m$ steps, the first of them is made terminal or phase 4 follows if there is no such word.

Let $v_0, v_1, ...$ be all words from $L(A)$ in order given by $\prec_A$. Now we use induction to prove that $L(G) \supseteq L(A)$.

Base: $\sigma\$ \Rightarrow_G^* v_0\$$:
The initial generative step is $\sigma \Rightarrow_G P_3H\overline{\epsilon}S\$$. Let us denote $m$ the current number of nonterminals A in the sentential form before the first S or $\frac{B}{B}$ symbol and $m_{v_0}$ is the minimal number of steps on which TM A accepts word $v_0$ (as defined in 4.1). In this case $m = 0$. From the assumptions 4.3 and 4.4 we have that while $m < m_{v_0}$ phase 4 follows after the phase 3 and then $m$ is incremented by one. Thus G on the sentential form $P_3H\overline{\epsilon}S\$$ repeats phases 3,4 and 2 until $m = m_{v_0}$. After this cycle sentential form $P_3A^{m_{v_0}}H\overline{\epsilon}S...\overline{a_k^{m_{v_0}}}S\$$ is derived. From the asumptions on TM A and from the fact that it accepts $v_0$ after $m_{v_0}$ steps we have that $v_0 \in \{\epsilon, a_1, ..., a_k^{m_{v_0}}\}$ thus the sentential form contains the subword $\overline{v_0}$. Furthermore, from minimality of $v_0$ in the sense of $\prec_A$ we have that $\forall w \in L(A)$ such that $w$ precedes $v_0$ in rlo we have that $m_w > m_{v_0}$ holds. Thus from the assumption 4.3 we have that for $m = m_{v_0}$:

$$P_3A^mH\overline{\epsilon}S...\overline{a_k^m}S\$ \Rightarrow_G^* v_0\$.$$

Inductive step: $\sigma \Rightarrow_G^* v_i\$$ implies $\sigma \Rightarrow_G^* v_{i+1}\$$:
we find the derivation $v_i\$ \Rightarrow_G^* v_{i+1}\$$. From the assumptions 4.1 and 4.2 we have that

$$v_i\$ \Rightarrow_G^* P_3A^{m_{v_i}}H\overline{w}S...S\overline{a_k^{m_{v_i}}}S\$,$$

where $w$ is the first succesor of $v_i$ in rlo. Let $d = m_{v_{i+1}} - m_{v_i}$. From the previously mentioned sentential form $G$ cycles $d$ times in the phases 3,4 and 2. The fact that phase

3 results $d$ times in the phase 4 follows from the order given by $\prec_A$. If some terminal word $w'$ would be derived after less than $d$ such cycles we would have a contradiction $m_{w'} < m_{v_{i+1}}$ and $v_{i+1} \prec_A w'$. During this process the number of nonterminals A is continually increased in phase 4 (assumption 4.5) so after $d$ repetitions of this cycle the sentential form

$$P_3 A^m H \overline{u} S ... S \overline{a_k^m} S\$$$

is generated, where $m = m_{v_i} + d$, $u = \epsilon$ if $d > 0$ and $u = w$ otherwise. In both cases $v_{i+1} \in \{u, ..., a_k^m\}$ from the definition of $\prec_A$. Furthermore, $v_{i+1}$ is the first of these words (in rlo) that TM A accepts on $m$ steps. Thus from the assumption 4.3 we have

$$P_3 A^m H \overline{u} S ... S \overline{a_k^m} S\$ \Rightarrow_G^* v_{i+1}\$.$$

$L(G) \subseteq L$

$w \in L(G)$ implies $\overline{w}\$ \Rightarrow_{G_A} w\$$. This implication follows from the fact, that terminal words are derived only in the phase 3 in which $H_{G_A}$ is used to simulate TM A on certain words. Then from $\overline{w}\$ \Rightarrow_{G_A} w\$$ and Lemma 2.14 we have $w \in L(A) = L$.

Now we shall verify the assumptions 4.1,...,4.5 so in the following part one generative step on certain sentential forms is shown with the corresponding computation of M:

Phase 1:

1. $w_0\$ \Rightarrow_G P_1 \overline{w_0}\$$
   This is the initial step of phase 1 in which G prepares for simulation of TM A on $w_0$ in order to compute $m_{w_0}$.
   For $w_0 = \epsilon$ computation of M is $(q_0, \$, P_1 \frac{B}{B} q_{A,0} \frac{B}{B}\$, q_0)$ and for $w_0 = b_1...b_n$ where $b_1, ..., b_n \in T$ we have computation
   $$(q_0, b_1, P_1\frac{B}{B}q_{A,0}\frac{b_1}{b_1}, q_{P_1,1})(q_{P_1,1}, b_2, \frac{b_2}{b_2}, q_{P_1,1})...(q_{P_1,1}, b_n, \frac{b_n}{b_n}, q_{P_1,1})(q_{P_1,1}, \$, \frac{B}{B}\$, q_{P_1,1}).$$

2. $P_1 \frac{B}{B} q_{A,0} \frac{w_0}{w_0} \frac{B}{B}\$ \Rightarrow_G P_1 A w A\$$
   where $q_{A,0} \notin F_A$ and $\overline{w_0}\$ \Rightarrow_{G_A} w\$$. Note that word $w$ consists of double track symbols and one nonterminal of some state of A, its second track represents the content of the tape of A and its first track contains the word $w_0$. So $w$ stores the information about current configuration of A and word $w_0$ at the same time. In this generative step the initial step of TM A on the word $w_0$ is simulated assuming that $(q_{A,0}w_0)$ is not the accepting configuration of A. Furthermore, the counters of simulated steps (numbers of nonterminals $A$ at the beginning and at the end of the sentential form) are incremented.
   Computation of M:
   $$(q_0, P_1, \epsilon, q_{P_1,2})(q_{P_1,2}, \frac{B}{B}, P_1, q_{P_1,7})(q_{P_1,7}, q_{A,0}, Av, p_A)h_1...h_l(q_A, \$, uA\$, q_{P_1,10})$$

for double track words $u, v$ and states $q_A, p_A \in K_{G_A}$ such that $([\frac{B}{B}], q_{A,0}, v, p_A)$, $(q_A, \$, u\$, p'_A)$, $h_1, ..., h_l \in H_{G_A}$ for some state $p'_A \in K_{G_A}$.

3. $P_1 A^m \frac{B}{B} w \frac{B}{B} A^m \$ \Rightarrow_G P_1 A^{m+1} \frac{B}{B} w' \frac{B}{B} A^{m+1} \$$
for $m > 0$, double track words $w, w'$ representing configurations of TM A, where it holds that $\frac{B}{B} w \frac{B}{B} \$ \Rightarrow_{G_A} \frac{B}{B} w' \frac{B}{B} \$$ and $z \notin F_A$ where $z$ is the first symbol of $w$ (in other words, configuration of A encoded in $w$ is not accepting). In this generative step, similarly to the previous one, one step of TM A from the given configuration is simulated and counters are incremented.
Computation of M:

$$(q_0, P_1, \epsilon, q_{P_1,2})(q_{P_1,2}, A, P_1, q_{P_1,4})(q_{P_1,4}, A, A, q_{P_1,4})^{m-1}$$
$$(q_{P_1,4}, \frac{B}{B}, \epsilon, q_{P_1,5})(q_{P_1,5}, z, AAv, q_{A,z})h_1...h_l$$
$$(q_A, A, uAA, q_{P_1,10})(q_{P_1,10}, A, A, q_{P_1,10})^{m-1}(q_{P_1,10}, \$, \$, q_{P_1,10})$$

where $([\frac{B}{B}], z, v, q_{A,z})$, $(q_A, \$, u, q)$, $h_1, ..., h_l \in H_{G_A}$ for some $q \in K_{G_A}$.

4. $P_1 A^m \frac{B}{B} q_F \frac{B^r w_0 B^s}{v_0} \frac{B}{B} A^m \$ \Rightarrow_G P_1 A^m MSHw_1SA^m \$$
where $w_0 \in T^*, v_0 \in (N \cup T)^* B^*, r, s \in \mathbb{N}$ and $w_1$ is the next successor of $w_0$ in rlo and $q_F \in F_A$. In this step the simulation of TM A is terminated, because it has reached the accepting configuration $(q_F v_0)$. Terminal word $w_0$ is restored from the first track and incremented in rlo. Also nonterminal $M$ is written to the output which is used as a flag that the simulation of A has ended.
For $m > 0$ the first part of the computation of M is:

$$(q_0, P_1, \epsilon, q_{P_1,2})(q_{P_1,2}, A, P_1, q_{P_1,4})(q_{P_1,4}, A, A, q_{P_1,4})^{m-1}$$
$$(q_{P_1,4}, \frac{B}{B}, \epsilon, q_{P_1,5})(q_{P_1,5}, q_F, AMSH, q_{P_1,6})(q_{P_1,6}, \frac{B}{x}, \epsilon, q_{P_1,6})^r$$

for any $x \in N \cup T$ and in the other case when $m = 0$ we have:

$$(q_0, P_1, \epsilon, q_{P_1,2})(q_{P_1,2}, \frac{B}{B}, P_1, q_{P_1,7})(q_{P_1,7}, q_F, MSH, q_{P_1,6})(q_{P_1,6}, \frac{B}{x}, \epsilon, q_{P_1,6})^r.$$

The rest of the computation is similar in both cases and it depends on $w_0$ which is incremented in the sense of rlo. First, let us assume that $w_0 = a_k^l a_i b_1 ... b_n$, where $i < k$ and $b_1, ..., b_n \in T$, then the computation continues with

$$(q_{P_1,6}, \frac{a_k}{x}, a_1, q_{P_1,6})^l (q_{P_1,6}, \frac{a_i}{x}, a_{i+1}, q_{P_1,11})(q_{P_1,11}, \frac{b_1}{x}, b_1, q_{P_1,11})...$$
$$(q_{P_1,11}, \frac{b_n}{x}, b_n, q_{P_1,11})(q_{P_1,11}, \frac{B}{x}, \epsilon, q_{P_1,11})^{s+1}).$$

If $m=0$, the computation ends with $(q_{P_1,11},\$,S\$,q_{P_1,11})$ or with

$$(q_{P_1,11},A,SA,q_{P_1,8})(q_{P_1,8},A,A,q_{P_1,8})^{m-1}(q_{P_1,8},\$,\$,q_{P_1,8})$$

otherwise. If $w_0=a_k^n$ does not contain other symbol than $a_k$ then the next successor of $w_0$ in rlo is $a_1^{n+1}$. In such case the computaion of M continues with

$$(q_{P_1,6},\frac{a_k}{x},a_1,q_{P_1,6})^n(q_{P_1,6},\frac{B}{x},\epsilon,q_{P_1,6})^{s+1}$$

and ends with $(q_{P_1,6},\$,a_1S\$,q_{P_1,6})$ if $m=0$ or with

$$(q_{P_1,6},A,a_1SA,q_{P_1,8})(q_{P_1,8},A,A,q_{P_1,8})^{m-1}(q_{P_1,8},\$,\$,q_{P_1,8})$$

otherwise.

5. $P_1A^rMA^sSHw_1SA^{r+s}\$ \Rightarrow_G P_1A^{r-1}MA^{s+1}SHw_1SA^{r+s}\$$
for any $r>0, s\in\mathbb{N}, w_1\in T^*$. In this generative step the nonterminal $M$ is shifted one symbol to the left and the rest is copied in the state $q_{P_1,3}$. Computation of M is:

$$(q_0,P_1,\epsilon,q_{P_1,2})(q_{P_1,2},A,P_1,q_{P_1,4})(q_{P_1,4},A,A,q_{P_1,4})^{r-1}$$
$$(q_{P_1,4},M,MA,q_{P_1,3})(q_{P_1,3},x,x,q_{P_1,3})...$$

6. $P_1MA^mSHw_1SA^m\$ \Rightarrow_G P_2A^mSHw_1SA^m\$$.
When nonterminal $M$ finally appears right next to $P_1$ in the sentential form, both are replaced by nonterminal $P_2$ and the next phase follows. Computation of M:

$$(q_0,P_1,\epsilon,q_{P_1,2})(q_{P_1,2},M,P_2,q_{P_1,3})(q_{P_1,3},x,x,q_{P_1,3})...$$

From the above analysis we have that for $w_0\in L_m(A)$ and its successor in rlo $w_1$ it holds:

$$\begin{aligned}
w_0\$ &\Rightarrow_G P_1\overline{w_0}\$ && 1\\
&\Rightarrow_G^m P_1A^m\frac{B}{B}q_F\frac{w_0}{v}\frac{B}{B}A^m\$ && 2,3\\
&\Rightarrow_G P_1A^mMSHw_1SA^m\$ && 4\\
&\Rightarrow_G^m P_1MA^mSHw_1SA^m\$ && 5\\
&\Rightarrow_G P_2A^mSHw_1SA^m\$ && 6
\end{aligned}$$

thus the assumption 4.1 holds.

Phase 2:

7. $P_2A^muHavSwA^m\$ \Rightarrow_G P_2A^muaHvSwaA^m\$$
for any $u \in (T \cup \{S\})^*$, $v, w \in T^*$ and $a \in T$. In this generative step is one terminal (determined by the position of $H$) copied to the place just before the ending counter and $H$ is shifted to the right. By series of such steps G copies whole words separated by $S$ nonterminals. Computation of M:

$$(q_0, P_2, \epsilon, q_{P_2,1})(q_{P_2,1}, A, P_2, [P_2, A, 1])...([P_2, x, 1], y, x, [P_2, y, 1])...$$
$$([P_2, z, 1], H, z, q_{P_2,2})(q_{P_2,2}, a, aH, q_{P_2,a})(q_{P_2,a}, b, b, q_{P_2,a})...$$
$$(q_{P_2,a}, A, aA, q_{P_2,13})(q_{P_2,13}, A, A, q_{P_2,13})^{m-1}(q_{P_2,13}, \$, \$, q_{P_2,13})$$

where $x, y, z \in T \cup \{S\}$, $a \in T$ and $b \in T \cup \{S\}$. We can see that G uses look ahead - set of states $[P_2, x, 1]$ - in order to be able to shift $M$ flag from the end to the beginning of the sentential form in future. After reading nonterminal $H$ and the following terminal $a$, the information about what symbol is to be copied is stored in the state $q_{P_2,a}$.

8. $P_2A^muHSa_k^la_ivA^m\$ \Rightarrow_G P_2A^muSWa_1^lCa_{i+1}vA^m\$$
where $u \in (T \cup \{A, S\})^*$ and $v \in T^*$. Nonterminal $H$ is not followed by terminal symbol like in the previous case, but by nonterminal $S$. It means that the whole terminal subword $a_k^la_iv$ was copied and its copy is now incremented in the sense of rlo. In order to do that, in some cases it is necessary to add one more terminal to the incremented word. In those cases then follows the check, whether the new subword is not longer than $m$ (if it is so, this phase shall be terminated in the following steps). Thus we want G to stop the process of copying terminals until it clarifies whether this phase shall end or continue and for that purpose is nonterminal $W$ written to the sentential form instead of $H$. In this particular case, after reading $...HSa_k^la_i$ it is clear that no additional terminal is needed to increment the copied subword so nonterminal $C$ is written to the output which indicates that process of copying may continue. Computation of M:

$$(q_0, P_2, \epsilon, q_{P_2,1})(q_{P_2,1}, A, P_2, [P_2, A, 1])...([P_2, y, 1], H, y, q_{P_2,2})$$
$$(q_{P_2,2}, S, SW, q_{P_2,3})(q_{P_2,3}, a_k, a_1, q_{P_2,3})^l(q_{P_2,3}, a_i, Ca_{i+1}, q_{P_2,5})$$
$$(q_{P_2,5}, a, a, q_{P_2,5})...(q_{P_2,5}, A, A, q_{P_2,5})^{m'}(q_{P_2,5}, \$, \$, q_{P_2,5})$$

for any $y \in T \cup \{A, S\}$, $a \in T$.

9. $P_2A^muHSa_k^lA^m\$ \Rightarrow P_2A^muSWa_1^{l+1}IA^mI\$$. This generative step is similar to the previous one except that in this case successor in rlo of the copied word $a_k^l$ is the word $a_1^{l+1}$ thus G has to check whether $l + 1 \leq m$ or not. This is obtained by

writing nonterminals $I$ to the suggested positions (in this generative step) and their simultaneous shifting to the left in the next steps. Computation of M:

$$(q_0, P_2, \epsilon, q_{P_2,1})(q_{P_2,1}, A, P_2, [P_2, A, 1])...([P_2, y, 1], H, y, q_{P_2,2})(q_{P_2,2}, S, SW, q_{P_2,3})$$
$$(q_{P_2,3}, a_k, a_1, q_{P_2,3})^l(q_{P_2,3}, A, a_1 I, q_{P_2,4})(q_{P_2,4}, A, A, q_{P_2,4})^{m'-1}(q_{P_2,4}, \$, I\$, q_{P_2,4})$$

where $y \in T \cup \{A, S\}$.

10. $P_2 A^m uWvaCw \Rightarrow_G P_2 A^m uWvCaw$
where $u \in (T \cup \{S\})^*$, $v \in T^*$, $w \in (N \cup T)^*$ and $a \in T$. In this generative step the nonterminal $C$ is shifted to the left. For that purpose look ahead is provided by the set of states $[P_2, b, 2], \forall b \in T$ used. Computation of M:

$$(q_0, P_2, \epsilon, q_{P_2,1})(q_{P_2,1}, A, P_2, [P_2, A, 1])...([P_2, y, 1], W, y, q_{P_2,6})(q_{P_2,6}, a, W, [P_2, a, 2])...$$
$$([P_2, b_1, 2], b_2, b_1, [P_2, b_2, 2])...([P_2, b_3, 2], C, Cb_3, q_{P_2,9})(q_{P_2,9}, x, x, q_{P_2,9})...$$

for any $y \in T \cup \{A, S\}$, $a, b_1, b_2, b_3 \in T$ and $x \in N \cup T$.

11. $P_2 A^m uWCvA^{m'}\$ \Rightarrow_G P_2 A^m uHvSA^{m'}\$$
where $u \in (T \cup \{S\}$ and $v \in T^*$. As we mentioned, nonterminal $C$ indicates that G shall continue in copying the last terminal subword in the sentential form. When it appears right after $W$ sentential form can be modified so that situation 7 occurs. Computation of M:

$$(q_0, P_2, \epsilon, q_{P_2,1})(q_{P_2,1}, A, P_2, [P_2, A, 1])...([P_2, y, 1], W, y, q_{P_2,6})(q_{P_2,6}, C, H, q_{P_2,8})$$
$$(q_{P_2,8}, a, a, q_{P_2,8})...(q_{P_2,8}, A, SA, q_{P_2,18})(q_{P_2,18}, x, x, q_{P_2,18})...$$

for any $y \in T \cup \{A, S\}$, $a \in T$ and $x \in N \cup T$.

12. $P_2 A^m uWvaIwA^r IA^s\$ \Rightarrow_G P_2 A^m uWvIawA^{r-1}IA^{s+1}\$$
for arbitrary $u \in (T \cup \{S\})^*$, $v, w \in T^*$, $a \in T$, and $r > 0$. Nonterminals $I$ are shifted to the left in order to compare the number of terminals after $W$ and number of nonterminals $A$ at the end of the sentential form. Computation of M:

$$(q_0, P_2, \epsilon, q_{P_2,1})(q_{P_2,1}, A, P_2, [P_2, A, 1])...([P_2, y, 1], W, y, q_{P_2,6})$$
$$(q_{P_2,6}, b_1, W, [P_2, b_1, 2])...([P_2, b_2, 2], I, Ib_2, q_{P_2,16})(q_{P_2,16}, b_3, \epsilon, [P_2, b_3, 4])$$
$$([P_2, b_3, 4], b_4, b_3, [P_2, b_4, 4])...([P_2, b_5, 4], A, b_5, q_{P_2,12})(q_{P_2,12}, A, A, q_{P_2,12})^{r-1}$$
$$(q_{P_2,12}, I, IA, q_{P_2,12})(q_{P_2,12}, A, A, q_{P_2,12})^s(q_{P_2,12}, \$, \$, q_{P_2,12})$$

where $y \in T \cup \{A, S\}$ and $b_1, b_2, b_3, b_4, b_5 \in T$.

13. $P_2A^muWvaIwbIA^m\$ \Rightarrow_G P_2A^muWvIawM\$$
for any $u \in (T \cup \{S\})^*$, $v, w \in T^*$ and $a, b \in T$. This is one possible outcome of comparison of $|vawb|$ and $|A^m|$. There is at least one terminal between $W$ and first $I$ but no nonterminal $A$ before the second $I$ thus $|vawb| > m$. This means that G shall move to the next phase so nonterminal $M$ is written to the output and counter $A^m$ is deleted. In the next steps, $M$ is shifted to the left deleting all symbols after $W$ (following case). Computation of M starts similarly to the previous case, the difference occurs when M reads the second nonterminal $I$ in the state $[P_2, b, 4]$:
$$...([P_2, b, 4], I, M, q_{P_2,11})(q_{P_2,11}, A, \epsilon, q_{P_2,11})^{m'}(q_{P_2,11}, \$, \$, q_{P_2,11}).$$

14. $P_2A^muWv_0aIv_1bMv_2\$ \Rightarrow_G P_2A^muWv_0aIv_1bMv_2\$$
for any $u \in (T \cup \{S\})^*$, $v_0, v_1, v_2 \in T^*$ and $a, b \in T$. Again, M works as in the previous two cases to the point when it reads $M$ in the state $[P_2, b, 4]$:
$$...([P_2, b, 4], M, M, q_{P_2,9})(q_{P_2,9}, x, x, q_{P_2,9})...$$
where $x \in N \cup T$.

15. $P_2A^muWIvzIA^l\$ \Rightarrow_G P_2A^muWvCzA^l\$$
for any $l \in \mathbb{N}, u \in (T \cup \{S\})^*$, $v \in (T \cup \{A\})^*$ and $z \in T \cup \{A\}$. This is the second possible outcome of checking the length of the last terminal subword. Fact, that the first $I$ follows right after $W$ and the second $I$ is still present in the sentential form implies that the length of the compared terminal word is at most $m$, because otherwise situation 13 would have occured after the first $m$ steps of the comparative process so there would be nonterminal $M$ instead of the second $I$ in the sentential form. So the result of the comparison is clear after reading the second nonterminal $I$ which is then replaced by indicator $C$ (and shifted by one). Computation of M:
$$(q_0, P_2, \epsilon, q_{P_2,1})(q_{P_2,1}, A, P_2, [P_2, A, 1])...([P_2, y_0, 1], W, y_0, q_{P_2,6})$$
$$(q_{P_2,6}, I, W, q_{P_2,7})(q_{P_2,7}, y_1, \epsilon, [P_2, a, 3])...([P_2, y_2, 3], y_3, y_2, [P_2, y_3, 3])$$
$$...([P_2, z, 3], I, Cz, q_{P_2,18})(q_{P_2,18}, A, A, q_{P_2,18})^l$$
where $y_0 \in T \cup \{S\}$, $y_1, y_2, y_3 \in T \cup \{A\}$.

Several cases of shifting nonterminal $M$ to the left according to content of the sentential form follows.

16. $P_2A^muWIvzM\$ \Rightarrow_G P_2A^muWvM\$$
for any $u \in (T \cup \{S\})^*$, $v \in (T \cup \{A\})^*$, $w \in (N \cup T)^*$ and $z \in T \cup \{A\}$. The computation of M is similar to the previous case except for the step on nonterminal $M$: $...([P_2, z, 3], M, M, q_{P_2,18})....$

17. $P_2A^muWvaM\$ \Rightarrow_G P_2A^muWvM\$$
for any $u \in (T\cup\{S\})^*$, $v \in T^*$, $a \in T$. Nonterminal $M$ is shifted to the left deleting terminals after $W$. Computation of M:

$$(q_0, P_2, \epsilon, q_{P_2,1})(q_{P_2,1}, A, P_2, [P_2, A, 1])...([P_2, y, 1], W, y, q_{P_2,6})$$
$$(q_{P_2,6}, b, W, [P_2, b, 2])...([P_2, a, 2], M, M, q_{P_2,9})(q_{P_2,9}, \$, \$, q_{P_2,9})$$

where $y \in T \cup \{S\}$, $b \in T$.

18. $P_2A^muWM\$ \Rightarrow_G P_2A^muM\$$
for any $u \in (T \cup \{S\})^*$ and $v \in (N \cup T)^*$. Computation of M:

$$(q_0, P_2, \epsilon, q_{P_2,1})(q_{P_2,1}, A, P_2, [P_2, A, 1])...([P_2, y, 1], W, y, q_{P_2,6})$$
$$(q_{P_2,6}, M, M, q_{P_2,9})(q_{P_2,9}, \$, \$, q_{P_2,9})$$

where $y \in T \cup \{S\}$.

19. $P_2uyMv\$ \Rightarrow_G P_2uMyv\$$
for any $v \in (N \cup T)^*$ and $u \in (T \cup \{A, S\})^*$, $y \in T \cup \{A, S\}$ such that either first symbol of $u$ is $A$ or $u = \epsilon, y = A$. Computation of M:

$$(q_0, P_2, \epsilon, q_{P_2,1})(q_{P_2,1}, A, P_2, [P_2, A, 1])...([P_2, y, 1], M, My, q_{P_2,9})(q_{P_2,9}, x, x, q_{P_2,9})...$$

where $x \in N \cup T$.

20. $P_2MA^mSw_1Sw_2...Sa_k^mS\$ \Rightarrow_G P_3A^mH\overline{w_1}S...\overline{a_k^m}S\$$
for any $w_1, ..., a_k^m \in T^*$. This is the final step of this phase, nonterminal $M$ follows right after $P_2$ so they can be both replaced by $P_3$. Furthermore, terminal subwords are converted to the form that computation of TM A can be simulated on them and nonterminal $H$ is written at the end of the counter. Computation of M:

$$(q_0, P_2, \epsilon, q_{P_2,1})(q_{P_2,1}, M, P_3, q_{P_2,14})(q_{P_2,14}, A, A, q_{P_2,14})...(q_{P_2,14}, S, H\frac{B}{B}q_{A,0}, q_{P_2,15})$$
$$(q_{P_2,15}, a, \frac{a}{a}, q_{P_2,15})...(q_{P_2,15}, S, \frac{B}{B}S, q_{P_2,10})(q_{P_2,10}, a, \frac{B}{B}q_{A,0}\frac{a}{a}, q_{P_2,15})...(q_{P_2,10}, \$, \$, q_{P_2,10})$$

where $a \in T$. Note that there may occur situation when M works on sentential form $P_2MA^mSa_k^m\$$ (with the only terminal subword and without nonterminal $S$ at the end). In such case the last step of the computation is $(q_{P_2,15}, \$, \frac{B}{B}S\$, q_{P_2,15})$ so the result would have the same structure as in originaly analyzed case.

We have that for any $w_1 = a_k^la_iv, l \in \mathbb{N}, i < k, v \in T^*$ :

$$P_2A^mSHw_1SA^m\$ \Rightarrow_G^* P_2A^mSw_1HSw_1A^m\$ \qquad 7$$

$$\Rightarrow_G P_2A^mSw_1SWa_k^lCa_{i+1}vA^m\$ \quad 8$$

$$\Rightarrow_G^* P_2A^mSw_1SWCa_k^la_{i+1}vA^m\$ \quad 10$$

$$\Rightarrow_G P_2A^mSw_1SHa_k^la_{i+1}vSA^m\$ \quad 11$$

$$\Rightarrow_G ... \Rightarrow_G$$

$$\Rightarrow_G P_2A^mSw_1...a_k^rSWa_1^{r+1}IA^mI\$ \quad 9$$

$$\Rightarrow_G P_2A^mSw_1...SWa_1^rIa_1A^{m-1}IA\$ \quad 12$$

$$\Rightarrow_G^r P_2A^mSw_1...SWIa_1^{r+1}A^{m-(r+1)}IA^{r+1}\$ \quad 12$$

$$\Rightarrow_G P_2A^mSw_1...SWa_1^{r+1}A^{m-r-2}CA^{r+2}\$ \quad 15$$

$$\Rightarrow_G^{m-1} P_2A^mSw_1...SWCa_1^{r+1}A^m\$ \quad 10$$

$$\Rightarrow_G P_2A^mSw_1...SHa_1^{r+1}SA^m\$ \quad 11$$

$$\Rightarrow_G ... \Rightarrow_G$$

$$\Rightarrow_G P_2A^mSw_1...a_k^mSWa_1Ia_1^mIA^m\$$$

$$\Rightarrow_G P_2A^mSw_1...a_k^mSWIa_1^mM\$ \quad 13$$

$$\Rightarrow_G P_2A^mSw_1...a_k^mSWa_1^{m-1}M\$ \quad 16$$

$$\Rightarrow_G^{m-1} P_2A^mSw_1...a_k^mSWM\$ \quad 17$$

$$\Rightarrow_G P_2A^mSw_1...a_k^mSM\$ \quad 18$$

$$\Rightarrow_G^* P_2MA^mSw_1...a_k^mS\$ \quad 19$$

$$\Rightarrow_G P_3A^mH\overline{w_1}S...\overline{a_k^m}S\$ \quad 20$$

Note that for $w_1 = a_k^l$ or $|w_1| \geq m$ some of the above steps are skipped, but the result remains the same. Thus assumption 4.2 holds.

Phase 3:

21. $P_3A^rHA^s\frac{B}{B}z_1v_1\frac{B}{B}S\frac{B}{B}...\frac{B}{B}z_lv_l\frac{B}{B}S\$ \Rightarrow_G P_3A^{r-1}HA^{s+1}v_1'...v_l'\$$
for any $r > 0$ and $z_iv_i \in \left(\frac{\Gamma_A}{\Gamma_A}\right)^* K_A \left(\frac{\Gamma_A}{\Gamma_A}\right)^*$; $v_i' \in (\frac{B}{B}\left(\frac{\Gamma_A}{\Gamma_A}\right)^* K_A \left(\frac{\Gamma_A}{\Gamma_A}\right)^* \frac{B}{B}S) \cup \{\epsilon\}$ such that $v_i' = \epsilon \iff z_i \in F_A$ and $v_i' = u_iS$ otherwise, where $\frac{B}{B}z_iv_i\frac{B}{B}\$ \Rightarrow_{G_A} u_i\$$ for $i = 1, ..., l$. In other words, $z_iv_i$ stores some terminal word in its first track and configuration of TM A in its second track. In this generative step one step of TM A is simulated (using $G_A$ from Lemma 2.14) from those configurations. If any of those configurations is accepting, then the whole block separated by $S$ is deleted from the sentential form, because that means that corresponding word from the first track is accepted by A on less than $r+s$ steps. Nonterminal $H$ is used to count the number of steps that remain to simulate - it is the number of nonterminals $A$ between $P_3$ and $H$. Computation of M:

$$(q_0, P_3, \epsilon, q_{P_3,1})(q_{P_3,1}, A, P_3, q_{P_3,3})(q_{P_3,3}, A, A, q_{P_3,3})^{r-1}$$

$$(q_{P_3,3}, H, HA, q_{P_3,4})(q_{P_3,4}, A, A, q_{P_3,4})^s(q_{P_3,4}, \frac{B}{B}, \epsilon, q_{P_3,5}),$$

then $l$ cycles follow, each has one of two scenarios depending on $z_i$. For $z_i \in F_A$ we have:

$$(q_{P_3,5}, z_i, \epsilon, q_{P_3,6})(q_{P_3,6}, \frac{x}{y}, \epsilon, q_{P_3,6})...(q_{P_3,6}, S, \epsilon, q_{P_3,7})(q_{P_3,7}, \frac{B}{B}, \epsilon, q_{P_3,5})$$

and for $z_i \notin F_A$ we have

$$(q_{P_3,5}, z_i, v, q_{A,1})h_1...h_{|v_i|+1}(q_{A,|v_i|+1}, S, S, q_{P_3,7})(q_{P_3,7}, \frac{B}{B}, \epsilon, q_{P_3,5})$$

where $(\left[\frac{B}{B}\right], z_i, v, q_{A,1}), h_1, ..., h_{|v_i|+1} \in H_{G_A}$ and $q_{A,|v_i|+1} = pr_4(h_{|v_i|+1})$. The last cycle of the computatation remains uncompleted because sentential form ends by symbols $S\$$ thus $(q_{P_3,7}, \$, \$, q_{P_3,7})$ is the last step.

22. $P_3HA^m\frac{B}{B}z_1v_1\frac{B}{B}S...z_fv_f...\frac{B}{B}z_lv_l\frac{B}{B}S\$ \Rightarrow_G P_3WA^mCw_f\$$
for any $z_iv_i \in \left(\frac{\Gamma_A}{\Gamma_A}\right)^* K_A \left(\frac{\Gamma_A}{\Gamma_A}\right)^*$, $f \in \{1, ..., l\}$ where $w_f \in T^*$ is the word from the first track of $v_f$ and $z_f \in F_A, z_1, ..., z_{f-1} \notin F_A$. There is no nonterminal $A$ between $P_3$ and $H$ which means that $m$ steps of TM A were simulated so far. Thus G checks whether any of $l$ simulations ended in accepting configuration. Input word of the first such configuration is restored (from the first track) and the rest is deleted. The fact that there was an accepting configuration and some word is to be made terminal is indicated by the nonterminal $C$ in the sentential form. Computation of M:

$$(q_0, P_3, \epsilon, q_{P_3,1})(q_{P_3,1}, H, P_3W, q_{P_3,11})(q_{P_3,11}, A, A, q_{P_3,11})^m$$
$$(q_{P_3,11}, \frac{B}{B}, \epsilon, q_{P_3,12})(q_{P_3,12}, z_1, \epsilon, q_{P_3,15})(q_{P_3,15}, y, \epsilon, q_{P_3,15})...$$
$$(q_{P_3,15}, S, \epsilon, q_{P_3,16})(q_{P_3,16}, \frac{B}{B}, \epsilon, q_{P_3,12})...(q_{P_3,12}, z_f, C, q_{P_3,13})...$$
$$(q_{P_3,13}, \frac{a}{x_1}, a, q_{P_3,13})...(q_{P_3,13}, \frac{B}{x_2}, \epsilon, q_{P_3,13})...(q_{P_3,13}, S, \epsilon, q_{P_3,14})$$
$$(q_{P_3,14}, x_3, \epsilon, q_{P_3,14})...(q_{P_3,14}, \$, \$, q_{P_3,14})$$

where $y \in (\frac{\Gamma_A}{\Gamma_A}) \cup K_A$ and $a, x_1, x_2, x_3 \in T$. In case that $z_1, ..., z_l \notin F_A$ M does not reach the state $q_{P_3,13}$ during the computation so the last arc is $(q_{P_3,16}, \$, M\$, q_{P_3,16})$ and the output sentential form is $P_3WA^mM\$$, where the nonterminal $M$ indicates that no accepting configuration was reached so G is moving to the phase 4.

23. $P_3WA^mCw\$ \Rightarrow_G P_3WA^{m-1}Cw\$$
for any $w \in T^*, m > 0$. Computation of M:

$$(q_0, P_3, \epsilon, q_{P_3,1})(q_{P_3,1}, W, P_3, q_{P_3,8})(q_{P_3,8}, A, W, q_{P_3,17})(q_{P_3,17}, A, A, q_{P_3,17})^{m-1}$$

$$(q_{P_3,17}, C, C, q_{P_3,10})(q_{P_3,10}, a, a, q_{P_3,10})...(q_{P_3,10}, \$, \$, q_{P_3,10})$$

for any $a \in T$.

24. $P_3WCw\$ \Rightarrow P_3Cw\$$
for any $w \in T^*$. Computation of M:

$$(q_0, P_3, \epsilon, q_{P_3,1})(q_{P_3,1}, W, P_3, q_{P_3,8})(q_{P_3,8}, C, C, q_{P_3,10})(q_{P_3,10}, a, a, q_{P_3,10})...$$

for any $a \in T \cup \{\$\}$.

25. $P_3Cw\$ \Rightarrow_G w\$$
for any $w \in T^*$. Computation of M:

$$(q_0, P_3, \epsilon, q_{P_3,1})(q_{P_3,1}, C, \epsilon, q_{P_3,10})(q_{P_3,10}, a, a, q_{P_3,10})...$$

for any $a \in T \cup \{\$\}$.

26. $P_3WA^rMA^s\$ \Rightarrow_G P_3WA^{r-1}MA^{s+1}\$$
for any $r > 0$. Computation of M:

$$(q_0, P_3, \epsilon, q_{P_3,1})(q_{P_3,1}, W, P_3, q_{P_3,8})(q_{P_3,8}, A, W, q_{P_3,17})(q_{P_3,17}, A, A, q_{P_3,17})^{r-1}$$
$$(q_{P_3,17}, M, MA, q_{P_3,9})(q_{P_3,9}, A, A, q_{P_3,9})^s(q_{P_3,9}, \$, \$, q_{P_3,9}).$$

27. $P_3WMA^m\$ \Rightarrow_G P_3MA^m\$$.
Computation of M:

$$(q_0, P_3\epsilon, q_{P_3,1})(q_{P_3,1}, W, P_3, q_{P_3,8})(q_{P_3,8}, M, M, q_{P_3,9})$$
$$(q_{P_3,9}, A, A, q_{P_3,9})^m(q_{P_3,9}, \$, \$, q_{P_3,9}).$$

28. $P_3MA^m\$ \Rightarrow_G P_4A^m\$$.
Computation of M:

$$(q_0, P_3, \epsilon, q_{P_3,1})(q_{P_3,1}, M, P_4, q_{P_3,2})(q_{P_3,2}, A, A, q_{P_3,2})^m(q_{P_3,2}, \$, \$, q_{P_3,2}).$$

From the cases analyzed we have that:

$$P_3A^mH\overline{w_1}S...\overline{w_r}S\$ \Rightarrow_G^m P_3HA^mw_1'S...w_r'S\$ \qquad 21$$

where for all $i \in \{1, ..., r\}$: $w_i' = \epsilon$ for all $w_i$ which are accepted by TM A on less than $m$ steps and $\overline{w_i}\$ \Rightarrow_{G_A} w_i'\$$ otherwise. For all nonempty words $w_i$ let $z_i$ be their second symbol (the first is $\frac{B}{B}$). If there is a number $f \leq r$ such that $z_f \in F_A$, $z_j \notin F_A$ for $j < f$ then the derivation continues as follows

$$\begin{aligned} \ldots &\Rightarrow_G P_3 W A^m C w_f \$ && 22 \\ &\Rightarrow_G^{m+2} w_f \$ && 23, 24, 25 \end{aligned}$$

and the assumption 4.3 holds.

If there is no such $f$ then the next generative steps are

$$\begin{aligned} \ldots &\Rightarrow_G P_3 W A^m M \$ \\ &\Rightarrow_G^{m+2} P_4 A^m \$ && 26, 27, 28 \end{aligned}$$

and the assumption 4.4 holds.

Phase 4:

29. $P_4 A^m \$ \Rightarrow_G P_4 W A A L A^{2m-2} R A A \$$
for any $m > 0$. This is the initial step of phase 4. In this generative step $2m+2$ nonterminals $A$ are written to the output in order to build two $m+1$ long counters. In following steps, nonterminals $L, R$ are simultaneously shifted towards themselves until they "meet" in the middle of $A^{2m+2}$. Computation of M follows:

$$(q_0, P_4, \epsilon, q_{P_4,1})(q_{P_4,1}, A, P_4 W A A L, q_{P_4,2})(q_{P_4,2}, A, AA, q_{P_4,2})^{m-1}(q_{P_4,2}, \$, RAA\$, q_{P_4,2}).$$

30. $P_4 W A^r L A^s R A^r \$ \Rightarrow_G P_4 W A^{r+1} L A^{s-2} R A^{r+1} \$$
for any $s > 1, r > 0$. Note that $s$ is even, because initially $s = 2m - 2$ from 29 and in sequence of these steps it is iteratively decreased by 2 so case $s = 1$ cannot occur. Computation of M:

$$\begin{aligned} (q_0, P_4, \epsilon, q_{P_4,1})(q_{P_4,1}, W, P_4, q_{P_4,3})(q_{P_4,3}, A, W, q_{P_4,4})(q_{P_4,4}, A, A, q_{P_4,4})^{r-1} \\ (q_{P_4,4}, L, A, q_{P_4,5})(q_{P_4,5}, A, AL, q_{P_4,6})(q_{P_4,6}, A, \epsilon, q_{P_4,7})(q_{P_4,7}, A, A, q_{P_4,7})^{s-2} \\ (q_{P_4,7}, R, RA, q_{P_4,7})(q_{P_4,7}, A, A, q_{P_4,7})^r(q_{P_4,7}, \$, \$, q_{P_4,7}). \end{aligned}$$

31. $P_4 W A^{m+1} L R A^{m+1} \$ \Rightarrow_G P_4 W A^{m+1} C S H S A^{m+1} \$$
for any $m$. Fact, that nonterminal $R$ follows immediately after $L$ in the sentential form indicates that they separate the block of $A$s into two equal halves, so G may start to move to another phase. Thus $LR$ is replaced by $CSHS$ - $C$ indicates that suffix $A^{m+1} SHS A^{m+1} \$$ was successfully derived so this phase may end and it is shifted to the beginning of the sentential form in the next steps. In contrast, $SHS$

remains at its position in the following steps as a separator and it represents empty word $\epsilon$ as $w_1$ (from 4.2) in the following phase 2. Computation of M:

$$(q_0, P_4, \epsilon, q_{P_4,1})(q_{P_4,1}, W, P_4, q_{P_4,3})(q_{P_4,3}, A, W, q_{P_4,4})$$
$$(q_{P_4,4}, A, A, q_{P_4,4})^m(q_{P_4,4}, L, A, q_{P_4,5})(q_{P_4,5}, R, CSHS, q_{P_4,8})$$
$$(q_{P_4,8}, A, A, q_{P_4,8})^{m+1}(q_{P_4,8}, \$, \$, q_{P_4,8}).$$

32. $P_4WA^rCv \Rightarrow_G P_4WA^{r-1}CAv$
for any $r > 0$, $v \in N^*$. Computation of M:

$$(q_0, P_4, \epsilon, q_{P_4,1})(q_{P_4,1}, W, P_4, q_{P_4,3})(q_{P_4,3}, A, W, q_{P_4,4})$$
$$(q_{P_4,4}, A, A, q_{P_4,4})^{r-1}(q_{P_4,4}, C, CA, q_{P_4,9})(q_{P_4,9}, x, x, q_{P_4,9})...$$

where $x \in N$.

33. $P_4WCv \Rightarrow P_4Cv$
for any $v \in N^*$. Computation of M:

$$(q_0, P_4, \epsilon, q_{P_4,1})(q_{P_4,1}, W, P_4, q_{P_4,3})(q_{P_4,3}, C, q_{P_4,9})(q_{P_4,9}, x, x, q_{P_4,9})...$$

where $x \in N$.

34. $P_4Cv \Rightarrow_G P_2v$
for any $v \in N^*$. Computation of M:

$$(q_0, P_4, \epsilon, q_{P_4,1})(q_{P_4,1}, C, P_2, q_{P_4,9})(q_{P_4,9}, x, x, q_{P_4,9})...$$

where $x \in N$.

From the above analysis we have

$$\begin{aligned}
P_4A^m\$ &\Rightarrow_G P_4WAALA^{2m-2}RAA\$ && 29\\
&\Rightarrow_G^{2m-2} P_4WA^{m+1}LRA^{m+1}\$ && 30\\
&\Rightarrow_G P_4WA^{m+1}CSHSA^{m+1}\$ && 31\\
&\Rightarrow_G^{m+1} P_4WCA^{m+1}SHSA^{m+1}\$ && 32\\
&\Rightarrow_G P_4CA^{m+1}SHSA^{m+1}\$ && 33\\
&\Rightarrow_G P_2A^{m+1}SHSA^{m+1}\$ && 34
\end{aligned}$$

for any $m > 0$. In case that $m = 0$ phase 4 has only one generative step :

$$(q_0, P_4, \epsilon, q_{P_4,1})(q_{P_4,1}, \$, P_2ASHS\$, q_{P_4,1}).$$

Thus the assumption 4.5 holds.

We have verified all of the assumptions 4.1, 4.2, 4.3, 4.4 and 4.5. We have also proven that they imply $L(G) = L(A)$, so inclusion $\mathcal{L}_{\$DG} \supseteq \mathcal{L}_{RE}$ holds. In order to prove $\mathcal{L}_{RE} = \mathcal{L}_{\$DG}$ we should show also the opposite inclusion $\mathcal{L}_{RE} \supseteq \mathcal{L}_{\$DG}$ but it is not hard to see that for given \$dgs G we can construct TM A which simulates G and checks whether G derived the given word so we leave formal proof to the reader. $\square$

# Chapter 5

# Measuring nondeterminism in g-systems

From the previous research [2] we know that one nondeterministic state and $|\Sigma| + 1$ nondeterministic arcs are sufficient to simulate any 1-a-transducer working over an alphabet $\Sigma$. The mentioned study suggests that some computational measure of nondeterminism in g-systems would be useful, because the previous results do not tell us how often the nondeterminism is used during the generative process. In this section we define and study such measures.

We introduce definition of the mapping from the previous work [2] which tells us how many decisions can be made from a given state on a given letter.

**Definition 5.1** *Let us consider arbitrary 1-a-transducer $M = (K, \Sigma, \Sigma, H, q_0, F)$ and a g-system $G = (N, T, M, \sigma)$. We define mapping dec from pairs (state, symbol) to integers as follows: for each state $q \in K$ and for each symbol $a \in \Sigma$:*

$$\begin{aligned}
&\textit{if there exists an arc } h \in H \textit{ such that } pr_1(h) = q \wedge pr_2(h) = a \textit{ then}:\\
&\qquad dec(q, a) = |\{h \in H \mid pr_1(h) = q \wedge pr_2(h) = a\}| - 1\\
&\textit{else}:\\
&\qquad dec(q, a) = 0.
\end{aligned}$$

## 5.1 Considering the number of generative steps

In this section, we define and study a metric which measures how many decisions can a given g-system make during a given number of generative steps.

**Definition 5.2** *Let $G = (N, T, M, \sigma)$ be a g-system, let $M = (K, N \cup T, N \cup T, H, q_0, F)$ be a 1-a-transducer. Let $\alpha(w) = (q_0, a_0, v_0, q_1)...(q_k, a_k, v_k, q_{k+1})$ be any computation of $M$ on $w = a_0...a_k$. We define mapping stepDec from computations of $M$ to integers as follows:*

$$stepDec(\alpha(w)) = \sum_{i=0}^{k} dec(q_i, a_i)$$

*We define mapping stepDec from sequences of computations of $M$ (corresponding to consecutive steps of derivation in a g-system) to integers for a sequence of computations $\alpha_1(w_1), ..., \alpha_m(w_m)$ such that $w_j = pr_3(\alpha_{j-1}(w_{j-1}))$ for $j \in \{2, ..., m\}$ as follows:*

$$stepDec(\alpha_1(w_1), ..., \alpha_m(w_m)) = \sum_{i=1}^{m} stepDec(\alpha_i(w_i))$$

*Finally, we define mapping stepDec from pairs of g-system and number of generative steps to integers as follows:*

$$stepDec(G, m) = max\,(stepDec(\alpha_1(\sigma), ..., \alpha_m(w_m)) \mid \alpha_1(\sigma)...\alpha_m(w_m) \in (\Pi_M)^m)$$

In what follows we search for an upper bound of the above defined metric in the general case - g-systems generating all recursively enumerable languages. First, we show that we can simulate an arbitrary \$dgs by an equivalent g-system G satisfying $stepDec(G, m) \leq m$ for all $m$ and then we prove that this upper bound can be iteratively improved.

A simulation of \$dgs is straightforward, nondeterminism is used just to decide whether to delete the endmarker. The only problem that could occur during the simulation is that our g-system could derive some "bad" words after deletion of the endmarker. To prevent this we use the following normal form:

**Definition 5.3** *Let $G = (N, T, M, \sigma)$ be an arbitrary deterministic g-system with endmarker, where $M = (K, N \cup T, N \cup T, H, q_0, F)$ is a deterministic 1-a-transducer. We say that G is in normal form if the following conditions hold:*

$$|F| = 1, \text{ and } q_0 \notin F \tag{5.1}$$

$$\text{for the only accepting state } q \in F, \forall h \in H: \; if \; pr_4(h) = q \text{ then } pr_2(h) = \$. \tag{5.2}$$

**Lemma 5.4** *For arbitrary \$dgs $G = (N, T, M, \sigma)$, there exists \$dgs $G'$ in normal form such that $L(G) = L(G')$.*

*Proof.* Let $M = (K, N \cup T, N \cup T, H, q_0, F)$. We construct $M' = (K', N \cup T, N \cup T, H', q_0', F')$ as follows:

$$K' = K \cup \{q_F, q_N\}$$
$$H_{\$old} = \{h \mid h \in H : pr_2(h) = \$\}$$
$$H_{\$new} = \{(q, \$, v, r) \mid \forall h = (q, \$, v, p) \in H_{\$old}, \quad where\ r = q_F\ iff\ p \in F\ and\ r = q_N\ otherwise\}$$
$$H' = (H \cup H_{\$new}) - H_{\$old}$$
$$F' = \{q_F\}$$
$$q_0' = q_0$$

It is not hard to see that $G' = (N, T, M', \sigma)$ is in normal form because its only accepting state is $q_F \neq q_0$ which satisfies the condition 5.2.

Proof of $L(G) = L(G')$ is trivial because in construction of $H'$ we changed only $pr_4$ of those arcs from $H$ which have to be the last arcs of the computation and for any $h_{new} = (q, \$, v, r) \in H_{\$new}, h_{old} = (q, \$, v, p) \in H_{\$old} : r \in F' \iff p \in F$. □

At this point we are ready to prove the upper bound mentioned.

**Lemma 5.5** *Let $L \in \mathcal{L}_{RE}$ be an arbitrary language. There exists g-system $G = (N, T, M, \sigma)$ such that $L(G) = L$ and $\forall m : stepDec(G, m) \leq m - 1$.*

*Proof.* From the Theorem 2 we have that there exists $\$dgs\ G' = (N', T, M', \sigma)$, where $M' = (K', N' \cup T, N' \cup T, H', q_0', F')$ such that $L(G') = L$ and from lemma 5.4 we can assume that it is in normal form. We construct $M = (K, N \cup T, N \cup T, H, q_0, F)$ as follows:

$$N = N' \cup \{\$\};\ K = K';\ q_0 = q_0'; F = F';$$
$$H = H' \cup \{h = (q, \$, v, p) \mid (q, \$, v\$, p) \in H'\}$$

First, we show that $L(G) = L(G')$ by proving both inclusions:

$L(G) \supseteq L(G')$:

Let $w \in L(G')$ so there exists an integer $k$ such that $\sigma \Rightarrow^k_{G'} w$. From $H \supset H'$ we have that $M$ is able to simulate any computation of $M'$. $G$ generates $w$ as follows: $M$ simulates $M'$ in each generative step except for the last arc $h' = (q, \$, v\$, p)$ in the $k^{th}$ generative step which is replaced by $h = (q, \$, v, p)$.

$L(G) \subseteq L(G')$:

First, we mention three important observations about the work of $G$:

Observation 1: $G$ works on a prefix $v \in ((N \cup T) - \{\$\})^*$ in the same way as $G'$, because we neither removed nor added any arc $h$ such that $pr_2(h) \neq \$$ in the above construction.

Observation 2: for all computational steps $h' = (p, \$, v\$, q) \in H'$ there are exactly two computational steps (with the same $pr_1$ and $pr_2$) in $H$ and those are $(p, \$, v\$, q)$ and $(p, \$, v, q)$.

Observation 3: if $G$ deletes \$ from the sentential form then the next generative step will end in non-accepting state and $M$ will halt. This statement holds because we did not break the condition 5.2 from the definition of the normal form in construction of $G$.

Now let $w \in L(G)$ so there exist $k$ and a sequence of generative steps in $G$ such that $\sigma \Rightarrow_G^k w$. From the observation 3 it is clear that $G$ must delete the \$ symbol at the very last arc of the $k^{th}$ generative step because otherwise it would halt. Until that point $G$ and $G'$ work identically (observation 1) and from the observation 2 we have that at this point of generative process $G'$ reads the symbol \$ so it derives $w\$$. Thus $w \in L(G')$ because $G'$ is a \$dgs.

In the second part of the proof we shall show that $\forall m : stepDec(G, m) \leq m - 1$. $G$ is constructed from $\$dgs$ and for no new arc holds that $pr_3(h)$ contains \$ so at any point of generative process of G the sentential form contains at most one \$ symbol and it is located at the end of the sentential form. All arcs that break determinism of G satisfy the property $pr_2(h) = \$$ so for the computation $\alpha_M(w\$)$ where $w \in ((N \cup T) - \{\$\})^*$ holds that $stepDec(\alpha_M(w\$)) = dec(q, \$) = 1$ where $q$ is the state in which $G$ reads \$ and $stepDec(\alpha_M(w)) = 0$. Furthermore, $stepDec(\alpha_M(\sigma)) = dec(q_0, \sigma) = 0$ thus for any $m$ statement $stepDec(G, m) \leq m - 1$ holds. □

From the definition of the $stepDec(G, m)$ mapping we can see that it is not restricted to derivation of some particular word but it only depends on the length of the generative process. We exploit this fact to add a large amount of deterministic generative steps into the derivation in order to make the value of $stepDec(G, m)$ small compared to $m$.

**Lemma 5.6** *Let G be an arbitrary g-system and let $stepDec(G, m) = f(m)$. There exists a g-system $G' = (N', T', M', \sigma')$ such that $L(G) = L(G')$ and for all $m$*

$$stepDec(G', m) \leq f(log^*(m)) + log^*(m).$$

*Proof.* From the given g-system $G = (N, T, M, \sigma)$ and its 1-a-transducer $M = (K, N \cup T, N \cup T, H, q_0, F)$ we construct $G' = (N', T', M', \sigma')$ as follows:

$N' = N \cup N_{new}$, *where* $N_{new} = \{\sigma', 1, 0, A, B, C\}$ *and* $N \cap N_{new} = \emptyset$,

$T' = T$

$M' = (K', N' \cup T', N' \cup T', H', q_0', F'), where:$

$K' = K \cup K_{new},\ where\ K_{new} = \{q_0', q_{i1}, q_{i0}, q_{r1}, q_{r0}, q_{rep}, q_2\}\ and\ K \cap K_{new} = \emptyset$

$H' = H \cup H_{new},\ where\ H_{new} = \{$

$$(q_0', \sigma', 0B\sigma, q_0'),$$
$$(q_0', 0, \epsilon, q_{i0}),$$
$$(q_0', 1, \epsilon, q_{i1}),$$
$$(q_0', A, \epsilon, q_0'),$$

$$\left.\begin{array}{l}(q_{i1}, 1, 0, q_{i1}),\\ (q_{i1}, 0, 0, q_{i0}),\\ (q_{r0}, 0, 0, q_{r0}),\\ (q_{r0}, 1, 0, q_{r1}),\\ (q_{i0}, 1, 1, q_{r1}),\\ (q_{i0}, 0, 1, q_{r0}),\\ (q_{r1}, 0, 1, q_{r0}),\\ (q_{r1}, 1, 1, q_{r1}),\end{array}\right\}\text{These arcs increment the counter}$$

$$\left.\begin{array}{l}(q_{i1}, B, C, q_{rep}),\\ (q_{i1}, B, A00B, q_0),\end{array}\right\}\text{counter overflow: terminate or simulate one step of G}$$

$$\left.\begin{array}{l}(q_{i0}, B, 1B, q_{rep}),\\ (q_{r1}, B, 1B, q_{rep}),\\ (q_{r0}, B, 0B, q_{rep}),\end{array}\right\}\text{counter does not overflow, the rest of the sentential form is copied}$$

$$\left.\begin{array}{l}(q_{i0}, A, A, q_2),\\ (q_{i1}, A, A, q_2),\\ (q_{r0}, A, A, q_2),\\ (q_{r1}, A, A, q_2),\\ (q_2, 0, 00, q_2),\\ (q_2, B, B, q_{rep}),\end{array}\right\}\text{A is present in the sentential form - counter is being extended}$$

$$\left.\begin{array}{l}(q_{r0}, C, C, q_{rep}),\\ (q_{i0}, C, C, q_{rep}),\\ (q_{i1}, C, C, q_{rep}),\\ (q_{r1}, C, C, q_{rep}),\\ (q'_0, C, \epsilon, q_{rep}),\end{array}\right\} \text{sentential form contains C: deleting the counter}$$

$$(q_{rep}, x, x, q_{rep}) \quad \forall x \in N' \cup T'\}$$

$$F' = F \cup \{q'_0, q_{rep}\}$$

The part of $G'$ corresponding to $H_{new}$ is depicted in figure 5.1. Before we get to the proof, let us analyze the work of G':
G' maintains the binary counter at the beginning of the sentential form followed by the nonterminal B. In each generative step G' increments the counter by one (in reversal lexicographic order). If the counter overflows then G' has two options: it either simulates one generative step of G and extends the counter or it decides to terminate and deletes the counter. In the first case we have $1^k Bv \Rightarrow_{G'} 0^{k-1}A00Bv; v \in (N \cup T)^*$ and then follows $k-1$ steps in which $A$ is shifted to he left and number of nonterminals 0 after $A$ is doubled. Thus we have that $0^{k-1}A00Bv \Rightarrow_{G'}^{k-1} 0^{2^k}Bv$ so the length of the counter is exponentiated.

Corectness: $L(G') \subseteq L(G)$: let $w \in L(G')$ be a word. There exists a derivation of w in G'. As we can see from the construction of G', the important generative steps are those in which its counter overflows. Let $k$ be the number of such generative steps during the derivation of $w$. Thus in the first $k-1$ of these steps G' uses arc $(q_{i1}, B, 00B, q_0)$ after reading and incrementing the counter and then from $q_0$ it uses the arcs from $H$ in the rest of such generative step. In the $k^{th}$ important step G' uses arc $(q_{i1}, B, C, q_{rep})$ and the rest of the sentential form is copied. In the following generative steps G' does not change, add or delete any symbol after the nonterminal $C$. The counter does not contain any terminal symbol and in the very first generative step G' uses the arc $(q'_0, \sigma', 0B\sigma, q'_0)$ thus G derives $w$ after $k-1$ generative steps from the initial nonterminal $\sigma$.

$L(G') \supseteq L(G)$: let $w \in L(G)$ be a word so there exists derivation of $w$ in G: $\sigma \Rightarrow_G^k w$ for some $k$. We find a derivation of $w$ in G'. Again, the important generative steps of $G'$ are those in which the counter overflows because otherwise the symbols, which follow after nonterminal $B$ in the sentential form are just copied in the state $q_{rep}$. Let us assume a derivation of G' in which the arc $(q_{i1}, B, 00B, q_0)$ is used in the first $k$ such steps. Thus $G'$ simulates $G$ on the suffix following after nonterminal $B$ in the sentential form, because from the state $q_0$ it uses only arcs from $H$. So after the $k^{th}$ important step the sentential form $0^kBw$ is derived, because in the initial generative step $0B\sigma$ is derived from $\sigma'$. If

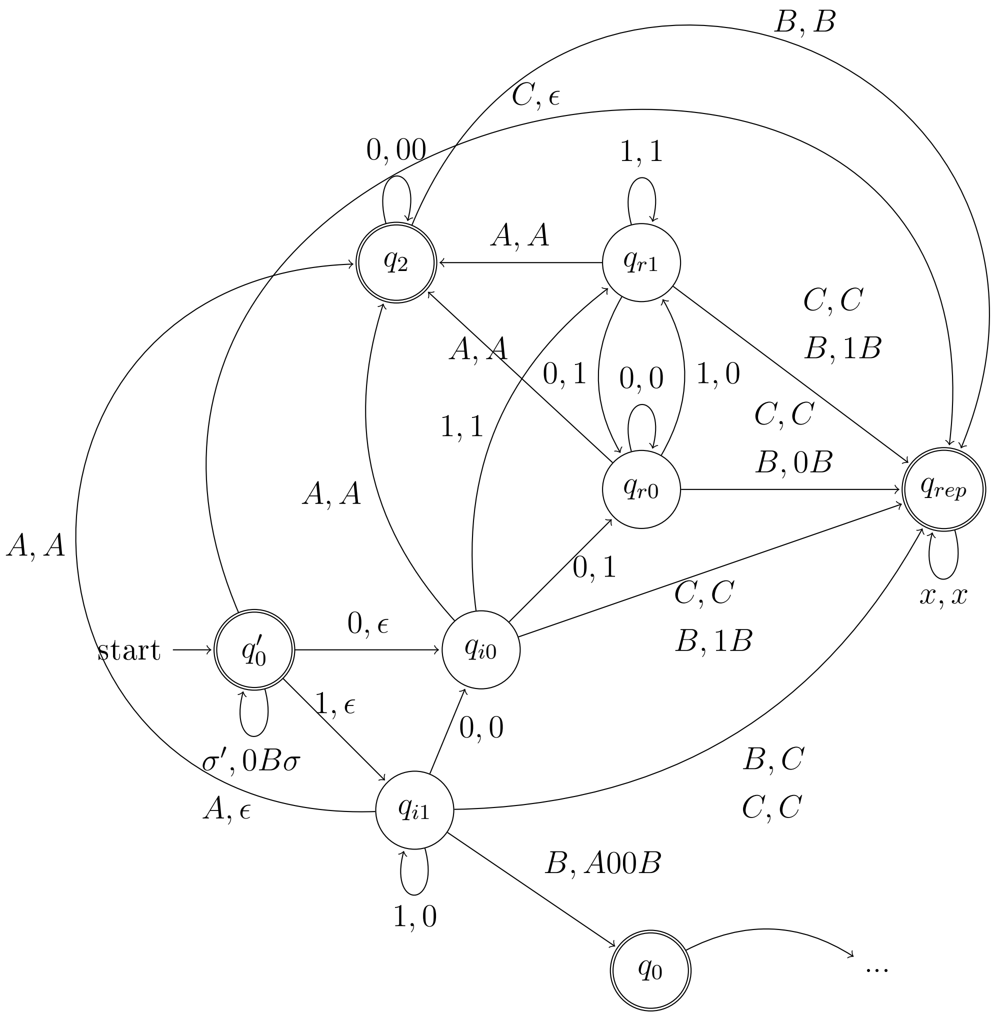


Figure 5.1: Part of M' which is responsible for making $\left.2^{2^{\cdot^{\cdot^{\cdot^{2}}}}}\right\}m$ purely deterministic generative steps for m simulated steps of G.

in the next important generative step G' uses the arc $(q_{i1}, B, C, q_{rep})$ then the subword $w$ is copied. G' does not modify any symbol after $C$ because for all arcs $h \in H'$ such that $pr_2(h) = C$ it holds that $pr_4 = q_{rep}$ which is a "copy" state. The only thing that remains to prove is that $G'$ deletes the counter properly. It is not hard to see that M' shorten the counter by one digit per generative step - it writes the $\epsilon$ in the initial step of the computation, one digit per any following 0 or 1 and then copies $C$ moving to the $q_{rep}$. After deleting the whole counter, the sentential form $Cw$ is derived and the following generative step starts by the arc $(q'_0, C, \epsilon, q_{rep})$ and then $w$ is copied.

Now let us count the number of deterministic and nondeterministic steps of G' during the generative process. The only nondeterminism occurs when the counter overflows so we have to answer the question how often it happens. The part of H' which increments the counter is deterministic so it is not hard to see that G' generates all binary numbers of the length of the counter (in rlo) before it overflows. Thus on counter $0^k$ G' does $2^k$ deterministic steps until it owerflows. As we mentioned, after nondeterministic step G' either terminates or the length of the counter is exponentiated - this can be done less than $log^*(m)$ times in $m$ generative steps. Thus we have that in $m$ generative steps of G' there are at most $log^*(m) + f(log^*(m))$ nondeterministic decisions used. □

**Theorem 3.** *For arbitrary language $L \in \mathcal{L}_{RE}$ there exists g-system $G$ such that $L(G) = L$ and $\forall m : stepDec(G, m) \leq 2log^*(m)$.*

*Proof.* This theorem follows from Lemmas 5.5 and 5.6. □

Note that we can use the concept from Lemma 5.6 iteratively to obtain arbitrary slow growth of the function $stepDec(G, m)$.

The only possible improvement from this result is a constant value of the stepDec() function, so our next question is whether there are languages in $\mathcal{L}_G - \mathcal{L}_{DG}$ which can be generated using constant number of nondeterministic decisions. The answer is yes and an example of such language is any $L \cup \{\epsilon\}$ where $L$ is an infinite language from $\mathcal{L}_{DG}$. It is not hard to see that only one nondeterministic step is sufficient in the very first generative step in which $G$ decides whether it generates $\epsilon$ or starts to generate $L$. There are also non-$\epsilon$ examples such as $L = \{a^n | n \geq 1\} \cup \{a^n b | n \geq 1\}$.

These examples bring us to a hypothesis that for $L = L_1 \cup L_2$ where $L_1, L_2 \in \mathcal{L}_{DG}$ we can construct a g-system $G$ such that $L(G) = L$ and $\exists c \in \mathbb{N} : \forall n : stepDec(G, n) \leq c$. Such statement can be easily proven for $L_1, L_2$ over disjoint alphabets, however, it does not seem to be true in general. The potential counterexample to this hypothesis can be the union of languages $L_1 = \{a^n b^n \mid n > 0\}, L_2 = \{a^{2n} b^n \mid n > 0\}$. Intuitively, a g-system generating the language $L_1 \cup L_2$ has to modify the generated words $a^{2i} b^{2i}$ and $a^{2i} b^i$ in a different way in order to derive the words from $L_1 \cup L_2$ in the following generative steps, but it cannot deterministically distinguish which of these two types of words is in the current sentential form.

We have that $stepDec(G, m)$ function splits the languages (corresponding g-systems by which they are generated) into two classes: those for which it is constant and those for which it is arbitrary slow increasing function. These results lead us to the idea of introducing another measure of nondeterminism in generative systems which would bring more granularity.

## 5.2 Considering the word length

The measure studied in this section is based on the minimal number of nondeterministic decisions needed to derive the given word or any word (from the generated language) of the given length. For this measure we obtain the linear dependence on the word length in general case and logarithmic for languages $\Sigma^*$ and unary languages. Furthermore, we show that for any recursive language $L$ over an alphabet $\Sigma$ this "nondeterministic" complexity depends on the ratio of the number of words of a given length $m$ to the number of words of length $m$ in $\Sigma^* - L$.

**Definition 5.7** *Let $G = (N, T, M, \sigma)$ be an arbitrary g-system, let $w$ be a word such that $w \in L(G)$ and let $n \in \mathbb{N}$. We define function lengthDec from pairs of g-system and word (pairs of g-system and integer resp.) to integers as follows:*

$$\begin{aligned} lengthDec(G, w) &= min(stepDec(\alpha_0(\sigma), ..., \alpha_m(w_m)) \mid pr_3(\alpha_m(w_m)) = w, \\ &\qquad \alpha_0(\sigma), ..., \alpha_m(w_m) \textit{ realize a derivation of } w \textit{ in } G) \\ lengthDec(G, n) &= max(lengthDec(G, w) \mid |w| = n) \end{aligned}$$

In other words, $lengthDec(G, w)$ is the smallest number of nondeterministic decisions needed to derive $w$ in $G$ and $lengthDec(G, n)$ is the number of decisions needed to derive any word in $L(G)$ of length $n$.

First, we investigate the upper bound for $lenghtDec(G, n)$ function in the general case - arbitrary recursively enumerable language. We use the fact that we can simulate any TM A on a given word by a deterministic g-system (Lemma 2.14). In the construction from the following theorem the g-system first nondeterministically derives an arbitrary word and then checks whether it belongs to $L(A)$.

**Theorem 4.** *Let $A$ be a Turing machine such that $L(A) \subseteq \Sigma^*$. There exists a g-system $G = (N, T, M, \sigma)$ such that $L(G) = L(A)$ and $lengthDec(G, n) \leq (n + 1)|\Sigma|$.*

*Proof.* We construct a g-system G which uses nondeterminism to derive $\overline{w}\$$ for any word $w$ over the alphabet $\Sigma$ and then works just like the \$dgs from Lemma 2.14 except it deletes the \$ symbol from the sentential form in case that A accepts $w$. We show only the part of set $H$ (from its 1-a-transducer) which is responsible for derivation of $\overline{w}\$$ from the initial nonterminal $\sigma$ because the rest is trivial:

$$\begin{aligned} H \supseteq \{&(q_0, \sigma, A\frac{B}{B}\$, q_1), \\ &(q_0, A, \frac{B}{B}q_A, q_1), \textit{where } q_A \textit{ is the initial state of } A \\ &(q_0, A, A\frac{a}{a}, q_1), \forall a \in \Sigma \\ &(q_1, \frac{c}{c}, \frac{c}{c}, q_1), \forall c \in \Sigma \cup \{B\} \\ &(q_1, \$, \$, q_1)\}. \end{aligned}$$

Note that nondeterminism is used only from the state $q_0$ on nonterminal $A$.

Corectness: we shall prove that G can generate sentential form $\frac{B}{B}q_A\frac{a_1}{a_1}...\frac{a_n}{a_n}\frac{B}{B}\$$ for any word $w = a_1...a_n \in \Sigma^*$. In the very first generative step G derives the sentential form $A\frac{B}{B}\$$. Let us assume the derivation of G in which the following $n+1$ generative steps are initialized by arcs:

$$(q_0, A, A\frac{a_n}{a_n}, q_1), ..., (q_0, A, A\frac{a_1}{a_1}, q_1), (q_0, A, \frac{B}{B}q_A, q_1).$$

Then we have that after $n+2$ generative steps G derives $\frac{B}{B}q_A\frac{a_1}{a_1}...\frac{a_n}{a_n}\frac{B}{B}\$$.

During this derivation G used nondeterminism $n+1$ times in the initial steps mentioned above and $dec(q_0, A) = |\Sigma|$ so we have that $lengthDec(G, n) = (n+1)|\Sigma|$ and the theorem follows. □

Although the construction from the previous theorem is straightforward, it seems that we can hardly obtain better result unless some different approach is used. We need to specify arbitrary word from $\Sigma^*$ on which G will simulate TM A. Thus for the given length $n$ we have $|\Sigma|^n$ possible words so $log(|\Sigma|^n) = |\Sigma|n$ decisions are needed.

Note that if we want to construct G which generates words (for example in rlo) from the previously derived terminal word then it has to find the end of the sentential form and mark it with a special symbol in order to simulate TM A on this word. That requires $log(n)$ nondeterministic decisions for all of $|\Sigma|^n$ words. Thus even assuming $L(G) \in \mathcal{L}_{REC}$ this approach would result in a bigger value of $lenghtDec(G, n)$ function than the construction from Theorem 4.

From the Theorem 1 we know that $\Sigma^+ \notin \mathcal{L}_{DG}$ for $|\Sigma| > 1$. However, we can construct dgs that generates all words from $\Sigma^n$ of any given length $n$. Thus in order to construct a g-system generating the language $\Sigma^*$ we need nondeterminism only for a derivation of arbitrarily long sentential form and the number of decisions sufficient to do so is logarithmic (to the derived length) as we prove in the following lemma:

**Lemma 5.8** *For an arbitrary alphabet* $\Sigma = \{a_0, ..., a_s\}$ *there exists g-system* $G = (N, T, P, \sigma)$ *such that* $L(G) = \Sigma^*$ *and* $lengthDec(G, n) \leq log(n)$.

*Proof.* We construct G as follows:

$$
\begin{aligned}
&T = \{a_0, ..., a_s\}\\
&N = \{A, \sigma\}\\
&M = (K, \Sigma, \Sigma, H, q_0, F) \textit{ where}\\
&K = \{q_0, q_1, q_2, q_3, q_4\}\\
&F = K\\
&H = \{\\
&\quad (q_0, \sigma, A, q_0),\\
&\quad (q_0, \sigma, \epsilon, q_0),\\
&\quad (q_0, A, a_0, q_1),\\
&\quad (q_0, A, AA, q_2),\\
&\quad (q_0, A, AAA, q_2),\\
&\quad (q_0, a_i, a_{i+1}, q_3)\ \forall i \in \{0, ..., s-1\},\\
&\quad (q_0, a_s, a_0, q_4),\\
&\quad (q_1, A, a_0, q_1),\\
&\quad (q_3, a_j, a_j, q_3)\ \forall j \in \{0, ..., s\},\\
&\quad (q_4, a_i, a_{i+1}, q_3)\ \forall i \in \{0, ..., s-1\},\\
&\quad (q_4, a_s, a_0, q_4)\}
\end{aligned}
$$

First, we shall prove the corectness of the construction: Inclusion $L(G) \subseteq \Sigma^*$ is trivial because $T = \Sigma$. The oposite inclusion, $L(G) \supseteq \Sigma^*$: Let $w = b_1...b_k \in \Sigma^*$ be an arbitrary word. We find the derivation of $w$ in $G$. If $w = \epsilon$ then $G$ can derive $w$ directly from $\sigma$ by $(q_0, \sigma, \epsilon, q_0)$. So now let us assume $|w| > 0$. As we can see, $G$ works in two phases. First, it generates arbitrary number of nonterminals $A$ and then it converts them to terminals. In the second phase, it generates all words of the derived length in reversal lexicographic order. We used similar approach of generating words in rlo earlier so we do not prove the corectness of this concept again. Now we shall prove that $G$ can derive the sentential form which contains exactly $k$ nonterminals $A$. Let

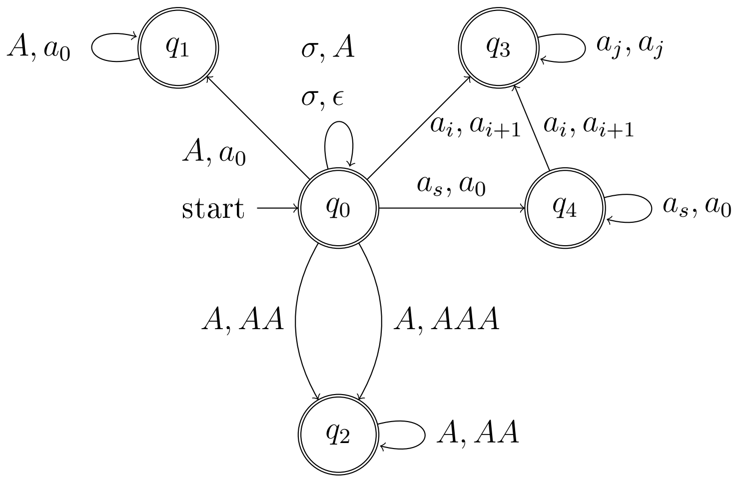


Figure 5.2: G-system that can derive any $w \in \Sigma^*, |w| = n$ using $log(n)$ nondeterministic decisions.

us denote $k_1, ..., k_m$ integers such that $k = 2^{k_1} + ... + 2^{k_m}$ and $k_1 > ... > k_m$. Let $l_i = k_1 - k_{i+1}$ for $i = 1, .., m-1$. The only nondeterministic state of G is $q_0$ so we can focus just on the first step of the computation in each generative step, because $\forall h \in H : pr_2(h) \neq \sigma \implies pr_4(h) \neq q_0$. Let us consider a derivation of G in which these first computational steps are $h_0, ..., h_{k_1}$ where $h_0 = (q_0, \sigma, A, q_0)$, $h_{l_i} = (q_0, A, AAA, q_2)$ and $h_j = (q_0, A, AA, q_2)$ for all $i \in \{1, ..., m-1\}, j \in \{1, ..., k_1\} - \{l_1, ..., l_{m-1}\}$. Let us count the number of nonterminals $A$ in the sentential form after $k_1$ generative steps: for the generative steps starting with $h_{l_i}$ the number of $A$ is doubled and for those starting with $h_j$ it is doubled and one more is added. Thus we have

$$\begin{aligned}\#A &= (...((2^{k_1-k_2} + 1)2^{k_2-k_3} + 1)...2^{k_{m-1}-k_m} + 1)2^{k_m} = \\ &= 2^{k_1-k_2+k_2-k_3...+k_{m-1}-k_m+k_m} + 2^{k_2-k_3+...+k_m} + ... + 2^{k_m} = 2^{k_1} + 2^{k_2} + ... + 2^{k_m} = k\end{aligned}$$

and in the next generative step $G$ begins with $h = (q_0, A, a_0, q_1)$. As we can see G used $k_1$ nondeterministic decisions during the generation and the length of the derived word is at least $2^{k_1}$ thus the lemma holds. □

Other languages for which nondeterminism is needed only to derive a sentential form of arbitrary length are unary languages.

**Lemma 5.9** *Let $L \in \mathcal{L}_{RE}$ be an arbitrary language over unary alphabet. There exists a g-system $G(N, T, M, \sigma)$ such that $L(G) = L$ and $lengthDec(G, n) \leq log(n)$.*

*Proof.* We construct G for a given Turing machine $A$ such that $L(A) \subseteq (\{a\})^*$. G simulates the computation of A on a word $a^n$ from the sentential form $\frac{B}{B} q_A \frac{a^n}{a^n} \frac{B}{B}\$$ similarly to the construction from Theorem 4 so we show only the part of $H$ (from M) which is responsible for the derivation of the sentential form $\frac{B}{B} q_A \frac{a^n}{a^n} \frac{B}{B}\$$, where $q_A$ is the initial state of $A$, for any $n$ .

$$
\begin{aligned}
H = \{ & (q_0, \sigma, A\frac{B}{B}\$, q_0), \quad \text{the initial arc} \\
& \left.\begin{array}{l} (q_0, A, AAA, q_1), \\ (q_0, A, AA, q_1), \\ (q_0, A, \frac{B}{B} q_A, q_2), \end{array}\right\} \quad \text{derive arbitrary number of nonterminals A} \\
& (q_1, A, AA, q_1), \\
& (q_2, A, \frac{a}{a}, q_2), \\
& (q_1, \frac{B}{B}, \frac{B}{B}, q_1), \\
& (q_1, \$, \$, q_1), \\
& (q_2, \frac{B}{B}, \frac{B}{B}, q_2), \\
& (q_2, \$, \$, q_2)\}
\end{aligned}
$$

The construction is similar to the one in Lemma 5.8 as well as its proof. □

In the proof of Lemma 5.9 and Theorem 4 we used a g-system which simulated the given TM A on a nondeterministically derived word. It is not hard to see that simulation of an arbitrary TM A on more words at once is meaningless because A may not halt on certain inputs. But situation is different assuming a language $L(A) \in \mathcal{L}_{REC}$. We can construct g-system G which simulates A on every word from $\Sigma^n$ (where $n$ is nondeterministically derived) and then makes one of the accepted words terminal. In general, we do not get better result than in Theorem 4 because G has to use nondeterminism to specify one from at most $|\Sigma|^n$ accepted words. But this approach can be usefull if the number of accepted words is considerably less as we show in the following theorem:

**Theorem 5.** *Let $L \in \mathcal{L}_{REC}$ be an arbitrary language. There exists a g-system G such that $L(G) = L$ and $lengthDec(G, n) \leq log(n) + log(|W_n|)$ where $W_n = \{w | w \in L, |w| = n\}$.*

*Proof.* We construct G for the given TM A which halts on every input such that $L(A) \subseteq \Sigma^*$, where $\Sigma = \{a_1, ..., a_k\}$. The idea of the construction is that G simulates A on every word of length $n$ and then makes the $m^{th}$ accepted word terminal. Numbers $n, m$ are generated nondeterministically. We divide the work of G into 5 phases:

1. G uses nondeterminism to derive the sentential form $P_2 A^n SASA^n S$ for arbitrary $n$ from the initial nonterminal $\sigma$.

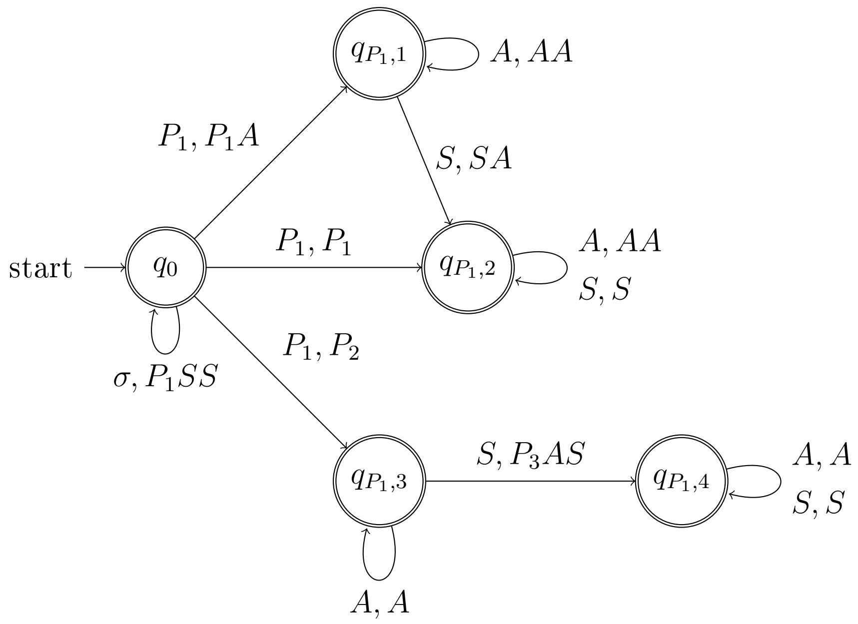


Figure 5.3: Phase 1: G uses nondeterminism to derive $\sigma \Rightarrow_G^* P_2A^nP_3ASA^nS$ for arbitrary $n$.

2. G is preparing to generate all words of length $n$ - the number of such words is $k^n$. After this phase the sentential form is $P_3A^{k^n}SA^nS$.
3. In this phase G iteratively copies the subword $A^nS$, converts this copy to $a_1^n$ and increments all previously generated and converted copies in rlo. The subword $A^{k^n}$ from the previous sentential form serves as a counter of such cycles. In the end G converts symbols to their double track version so after this phase the sentential form is $P_4C\overline{a_1^n}\$...\overline{a_k^n}\$$.
4. G nondeterministically generates subword $A^m$ for arbitrary $m$ at the beginning of the sentential form. This subword will be used to select the $m^{th}$ accepted word to be made terminal in the following phase.
5. G simulates TM A on each subword simultaneously. In each generative step it checks whether the first subword finished the simulation. If it is so, G either deletes it and decrements the counter if A accepts or just deletes it if A rejects. If A accepts and the counter is 0 then G makes the according subword (terminals from its first track) terminal and deletes anything else.

More formally, let $G' = (N', T', M', \sigma')$ be \$dgs from Lemma 2.14, where its 1-a-transducer $M' = (K', N' \cup T', N' \cup T', H', q_0', F')$. We construct $G = (N, T, M, \sigma)$ as follows:

$$N = N' \cup N_{new}, where\ N_{new} = \{\sigma, P_1, ..., P_5, A, S, C, W, L, R, \$\}\ and\ N' \cap N_{new} = \emptyset$$
$$T = T' = \{a_1, ..., a_k\}$$

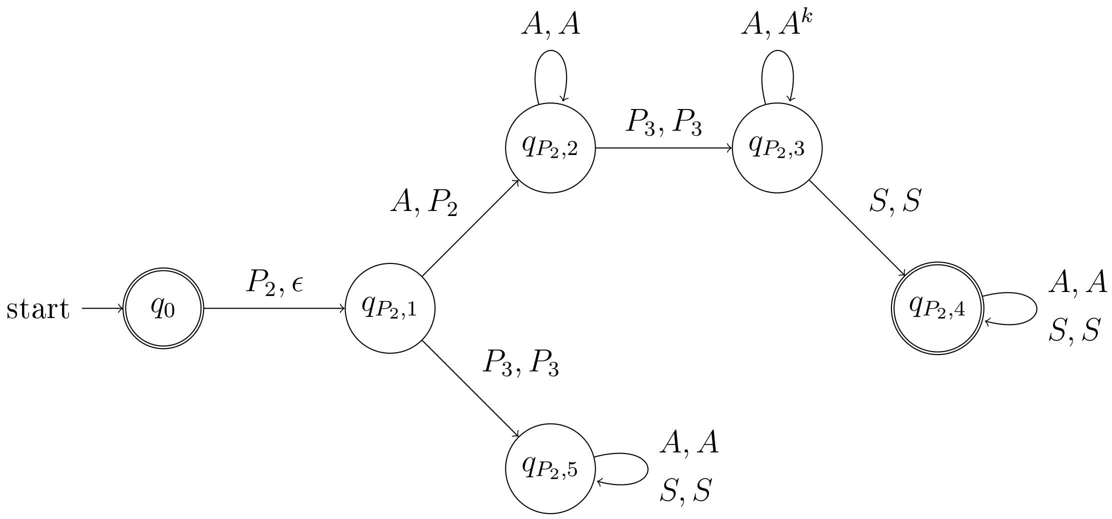


Figure 5.4: Phase 2: G deterministically derives $P_2A^nP_3ASA^nS \Rightarrow_G^* P_3A^{k^n}SA^nS$ for $k = |\Sigma|$.

$M = (K, N \cup T, N \cup T, H, q_0, F)$ *where* :

$K = K' \cup K_{new}$, *where* $K_{new} = \{q_0\} \cup \{q_{P_1,1}, ..., q_{P_1,4}\} \cup \{q_{P_2,1}, ..., q_{P_2,5}\} \cup$

$\cup \{q_{P_3,1}, ..., q_{P_3,19}\} \cup \{q_{P_4,1}, q_{P_4,2}\} \cup \{q_{P_5,1}, ..., q_{P_5,9}\}$ *and* $K' \cap K_{new} = \emptyset$

$F = K$

$H = H' \cup \{$

*phase* 1 :

$(q_0, \sigma, P_1SS, q_0)$,

$(q_0, P_1, P_1A, q_{P_1,1})$,

$(q_0, P_1, P_1, q_{P_1,2})$,

$(q_0, P_1, P_2, q_{P_1,3})$,

$(q_{P_1,1}, A, AA, q_{P_1,1})$,

$(q_{P_1,1}, S, SA, q_{P_1,2})$,

$(q_{P_1,2}, A, AA, q_{P_1,2})$,

$(q_{P_1,2}, S, S, q_{P_1,2})$,

$(q_{P_1,3}, A, A, q_{P_1,3})$,

$(q_{P_1,3}, S, P_3AS, q_{P_1,4})$,

$(q_{P_1,4}, A, A, q_{P_1,4})$,

$(q_{P_1,4}, S, S, q_{P_1,4})$,

*phase* 2 :

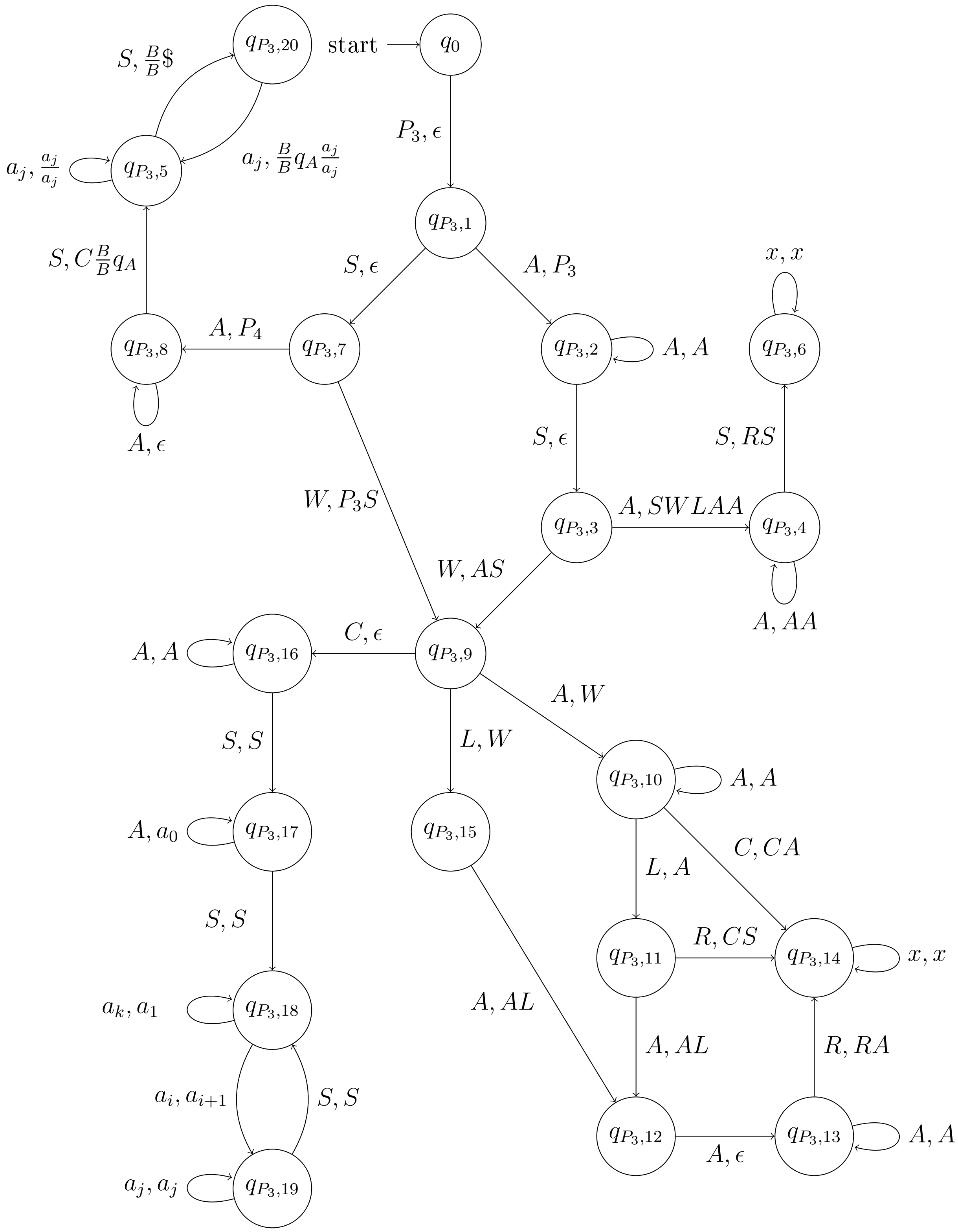


Figure 5.5: Phase 3: G derives all words from $\Sigma^n$ for a given $n$ and converts them to the double track version so TM A can be simulated on them. More precisely, this phase is responsible for the derivation of $P_3A^{k^n}SA^nS \Rightarrow_G^* P_4C\overline{a_1^n}\$\overline{a_2a_1^{n-1}}\$...\overline{a_k^n}\$$.

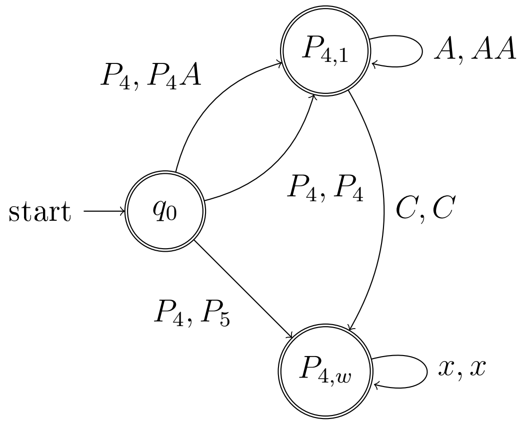


Figure 5.6: Phase 4: G uses nondeterminism to derive subword $A^m$ at the beginning of the sentential form for arbitrary $m$.

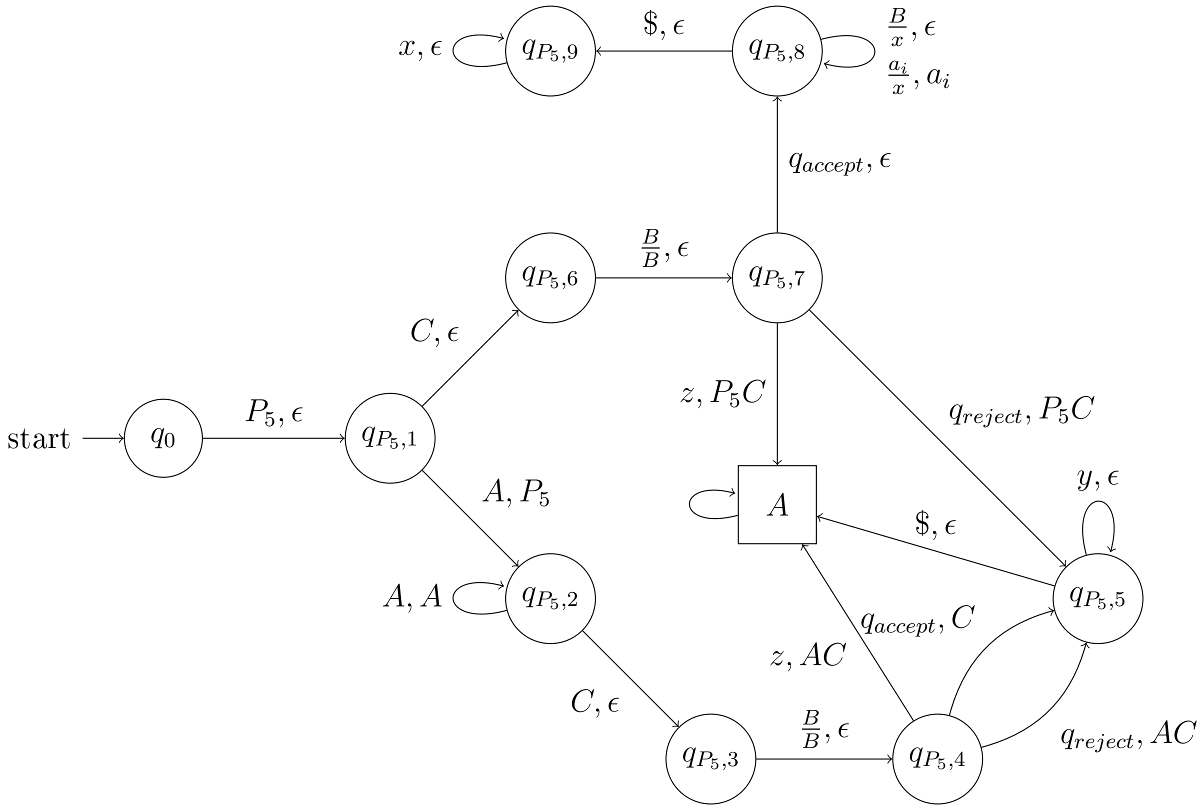


Figure 5.7: Phase 5: G simulates TM A on every word of length $n$ and then the $m^{th}$ of the accepted words is made terminal, all others are deleted.

$(q_{P_2,1}, A, P_2, q_{P_2,2}),$

$(q_{P_2,2}, A, A, q_{P_2,2}),$

$(q_{P_2,2}, P_3, P_3, q_{P_2,3}),$

$(q_{P_2,3}, A, A^k, q_{P_2,3}),$

$(q_{P_2,3}, S, S, q_{P_2,4}),$

$(q_{P_2,4}, A, A, q_{P_2,4}),$

$(q_{P_2,4}, S, S, q_{P_2,4}),$

$(q_{P_2,1}, P_3, P_3, q_{P_2,5}),$

$(q_{P_2,5}, S, S, q_{P_2,5}),$

$(q_{P_2,5}, A, A, q_{P_2,5}),$

$phase\ 3:$

$(q_0, P_3, \epsilon, q_{P_3,1}),$

$(q_{P_3,1}, A, P_3, q_{P_3,2}),$

$(q_{P_3,1}, S, \epsilon, q_{P_3,7}),$

$(q_{P_3,7}, A, P_4, q_{P_3,8}),$

$(q_{P_3,8}, A, \epsilon, q_{P_3,8}),$

$(q_{P_3,8}, S, C\dfrac{B}{B}q_A, q_{P_3,5}),$ where $q_A$ is the initial state of TM A

$(q_{P_3,7}, W, P_3S, q_{P_3,9}),$

$(q_{P_3,2}, S, \epsilon, q_{P_3,3}),$

$(q_{P_3,2}, A, A, q_{P_3,2}),$

$(q_{P_3,3}, A, SWLAA, q_{P_3,4}),$

$(q_{P_3,4}, A, AA, q_{P_3,4}),$

$(q_{P_3,4}, S, RS, q_{P_3,6}),$

$(q_{P_3,6}, x, x, q_{P_3,6}), \forall x \in (N \cup T)$

$(q_{P_3,3}, W, AS, q_{P_3,9}),$

$(q_{P_3,9}, A, W, q_{P_3,10}),$

$(q_{P_3,10}, A, A, q_{P_3,10}),$

$(q_{P_3,10}, L, A, q_{P_3,11}),$

$(q_{P_3,11}, A, AL, q_{P_3,12}),$

$(q_{P_3,12}, A, \epsilon, q_{P_3,13}),$

$(q_{P_3,13}, A, A, q_{P_3,13}),$

$(q_{P_3,13}, R, RA, q_{P_3,14})$,

$(q_{P_3,14}, x, x, q_{P_3,14}), \forall x \in (N \cup T)$

$(q_{P_3,10}, C, CA, q_{P_3,14})$,

$(q_{P_3,11}, R, CS, q_{P_3,14})$,

$(q_{P_3,9}, L, W, q_{P_3,15})$,

$(q_{P_3,15}, A, AL, q_{P_3,12})$,

$(q_{P_3,9}, C, \epsilon, q_{P_3,16})$,

$(q_{P_3,16}, A, A, q_{P_3,16})$,

$(q_{P_3,16}, S, S, q_{P_3,17})$,

$(q_{P_3,17}, A, a_1, q_{P_3,17})$,

$(q_{P_3,17}, S, S, q_{P_3,18})$,

$(q_{P_3,18}, a_k, a_1, q_{P_3,18})$,

$(q_{P_3,18}, a_i, a_{i+1}, q_{P_3,19})\ \forall i \in \{1, ..., k-1\}$,

$(q_{P_3,19}, S, S, q_{P_3,18})$,

$(q_{P_3,19}, a, a, q_{P_3,19})\ \forall a \in \Sigma$,

$(q_{P_3,5}, a, \frac{a}{a}, q_{P_3,5})\ \forall a \in \Sigma$,

$(q_{P_3,5}, S, \frac{B}{B}\$, q_{P_3,20})$,

$(q_{P_3,20}, a, \frac{B}{B} q_A \frac{a}{a}, q_{P_3,5})\ \forall a \in \Sigma$ where $q_A$ is the initial state of TM A,

$phase\ 4:$

$(q_0, P_4, P_4, q_{P_4,1})$,

$(q_0, P_4, P_5, q_{P_4,2})$,

$(q_{P_4,1}, A, AA, q_{P_4,1})$,

$(q_{P_4,1}, C, C, q_{P_4,2})$,

$(q_{P_4,2}, x, x, q_{P_4,2}), \forall x \in (N \cup T)$

$phase\ 5:$

$(q_{P_5,1}, A, P_5, q_{P_5,2})$,

$(q_{P_5,2}, A, A, q_{P_5,2})$,

$(q_{P_5,2}, C, \epsilon, q_{P_5,3})$,

$(q_{P_5,3}, \frac{B}{B}, \epsilon, q_{P_5,4})$,

$(q_{P_5,4}, q_{accept}, C, q_{P_5,5})$,

$(q_{P_5,4}, q_{reject}, AC, q_{P_5,5})$,

$(q_{P_5,4}, z, ACv, q)\ \forall z \in (N \cup T) - \{q_{accept}, q_{reject}\}, where\ (\begin{bmatrix} B \\ B \end{bmatrix}, z, v, q) \in H'$,

$(q_{P_5,5}, y, \epsilon, q_{P_5,5})\ \forall y \in (N \cup T) - \{\$\}$,

$(q_{P_5,5}, \$, \epsilon, q_0')$,

$(q_{P_5,1}, C, \epsilon, q_{P_5,6})$,

$(q_{P_5,6}, \frac{B}{B}, \epsilon, q_{P_5,7})$,

$(q_{P_5,7}, q_{reject}, P_5C, q_{P_5,5})$,

$(q_{P_5,7}, q_{accept}, \epsilon, q_{P_5,8})$,

$(q_{P_5,8}, \frac{B}{x}, \epsilon, q_{P_5,8})\ \forall x \in (N \cup T)$,

$(q_{P_5,8}, \frac{a_i}{x}, a_i, q_{P_5,8})\ \forall x \in (N \cup T), \forall i \in \{1, ..., k\}$

$(q_{P_5,8}, \$, \epsilon, q_{P_5,9})$,

$(q_{P_5,9}, x, \epsilon, q_{P_5,9})\ \forall x \in (N \cup T)$,

$(q_{P_5,7}, z, P_5Cv, q) \forall z \in (N \cup T) - \{q_{accept}, q_{reject}\}\ and\ (\begin{bmatrix} B \\ B \end{bmatrix}, z, v, q) \in H'\}$

The construction of each phase is shown in figures 5.3, 5.4, 5.5, 5.6, 5.7. Note that only phases 1 and 4 contain nondeterminism. We prove that in each phase G works as we suggested:

Phase 1: the proof that G can generate any number of nonterminals $A$ is similar to the one of Lemma 5.8. This phase ends after the generative step in which M uses the arc $(q_0, P_1, P_2, q_{P_1,3})$ in the first computational step. Now we show by induction that until that point after each generative step the sentential form is $P_1A^iSA^iS$ for some $i$. After the initial step the sentential form is $P_1S$ so the base of the induction holds. Let the sentential form be $v = P_1A^jSA^jS$. There are three possible computations of M on such input:

$$\begin{aligned} \alpha_1(v) =& (q_0, P_1, P_1A, q_{P_1,1})(q_{P_1,1}, A, AA, q_{P_1,1})^j(q_{P_1,1}, S, SA, q_{P_1,2}) \\ & (q_{P_1,2}, A, AA, q_{P_1,2})^j(q_{P_1,2}, S, S, q_{P_1,2}), \\ \alpha_2(v) =& (q_0, P_1, P_1, q_{P_1,2})(q_{P_1,2}, A, AA, q_{P_1,2})^j(q_{P_1,2}, S, S, q_{P_1,2}) \\ & (q_{P_1,2}, A, AA, q_{P_1,2})^j(q_{P_1,2}, S, S, q_{P_1,2}), \\ \alpha_3(v) =& (q_0, P_1, P_2, q_{P_1,3})(q_{P_1,3}, A, A, q_{P_1,3})^j(q_{P_1,3}, S, P_3AS, q_{P_1,4}) \\ & (q_{P_1,4}, A, A, q_{P_1,4})^j(q_{P_1,4}, S, S, q_{P_1,4}) \end{aligned}$$

We can easily see that $pr_3(\alpha_1(v)) = P_1A^{2j+1}SA^{2j+1}S$, $pr_3(\alpha_2(v)) = P_1A^{2j}SA^{2j}S$ and $pr_3(\alpha_3(v)) = P_2A^jP_3ASA^jS$.

Phase 2: We want to prove that $P_2A^nP_3ASA^nS \Rightarrow_G^{n+1} P_3A^{k^n}SA^nS$ for any positive integer $n$. In this phase the number of nonterminals $A$ between $P_3$ and $S$ is iteratively multiplied by $k$. The number of such steps is determined by the length of the first block of nonterminals $A$ in the sentential form which is decreased by one in each step. More precisely, on the sentential form $P_2A^iP_3A^jSA^mS$, for all $i, j > 0$, $m \geq 0$, the computation of M is as follows:

$$(q_0, P_2, \epsilon, q_{P_2,1})(q_{P_2,1}, A, P_2, q_{P_2,2})(q_{P_2,2}, A, A, q_{P_2,2})^{i-1}(q_{P_2,2}, P_3, P_3, q_{P_2,3})$$
$$(q_{P_2,3}, A, A^k, q_{P_2,3})^j(q_{P_2,3}, S, S, q_{P_2,4})(q_{P_2,3}, A, A, q_{P_2,4})^m(q_{P_2,4}, S, S, q_{P_2,4})$$

so we have that

$$P_2A^iP_3A^jSA^mS \Rightarrow_G P_2A^{i-1}P_3A^{jk}SA^mS.$$

On the prefix $P_2P_3$ the computation of M starts with $(q_0, P_2, \epsilon, q_{P_2,1})(q_{P_2,1}, P_3, P_3, q_{P_2,5})$ and the remaining nonterminals $A, S$ are copied in the state $q_{P_2,5}$ by the arcs $(q_{P_2,5}, A, A, q_{P_2,5})$ or $(q_{P_2,5}, S, S, q_{P_2,5})$ respectively. Thus we have that

$$\begin{aligned} P_2A^nP_3ASA^nS &\Rightarrow_G^n P_2P_3A^{k^n}SA^nS \\ &\Rightarrow_G P_3A^{k^n}SA^nS. \end{aligned}$$

Phase 3: We want to show that the derivation in this phase is the following:

$$P_3A^{k^n}SA^nS \Rightarrow_G^* P_4C\overline{a_1^n}\$\overline{a_2a_1^{n-1}}\$...\overline{a_k^n}\$.$$

First, we analyze particular generative steps of G and then we put them together obtaining the above derivation. In the following part we denote by $w_{rest}$ any word over $N \cup T$ and we use it in cases when M does not change the suffix of the input.

In what follows, the sequence of steps which is responsible for addition of the subword $a_1^n$ after the second nonterminal $S$ in the sentential form is analyzed. It consists of the following steps: block $A^n$ is doubled ($A^n \rightsquigarrow LA^{2n}R$), then using nonterminals $L, R$ its middle is found ($LA^{2n}R \rightsquigarrow A^nLRA^n$) and finally, the second block $A^n$ is converted to terminals ($A^nLRA^n \rightsquigarrow A^nSa_1^n$).

The computation of M on the sentential form $P_3A^iSA^nSw_{rest}$ for all $i, n > 0$, looks as follows:

$$(q_0, P_3, \epsilon, q_{P_3,1})(q_{P_3,1}, A, P_3, q_{P_3,2})(q_{P_3,2}, A, A, q_{P_3,2})^{i-1}(q_{P_3,2}, S, \epsilon, q_{P_3,3})$$
$$(q_{P_3,3}, A, SWLAA, q_{P_3,4})(q_{P_3,4}, A, AA, q_{P_3,4})^{n-1}(q_{P_3,4}, S, RS, q_{P_3,6})(q_{P_3,6}, x, x, q_{P_3,6})...$$

so the output is $P_3A^{i-1}SWLA^{2n}RSw_{rest}$. The nonterminal $W$ in the sentential form indicates that "subroutine" which adds the subword $a_1^n$ did not finish yet so it prevents the doubling of nonterminals $A$ while it is present.

Two cases in which nonterminals $L$ and $R$ are shifted towards themselves follow. Let us assume input $P_3A^iSWA^lLA^{2n-2l}RA^lSw_{rest}$ for any $i > 0$, $l < n$. If $l = 0$ M works as follows:

$$(q_0, P_3, \epsilon, q_{P_3,1})(q_{P_3,1}, A, P_3, q_{P_3,2})(q_{P_3,2}, A, A, q_{P_3,2})^{i-1}(q_{P_3,2}, S, \epsilon, q_{P_3,3})$$
$$(q_{P_3,3}, W, AS, q_{P_3,9})(q_{P_3,9}, L, W, q_{P_3,15})(q_{P_3,15}, A, AL, q_{P_3,12})(q_{P_3,12}, A, \epsilon, q_{P_3,13})$$
$$(q_{P_3,13}, A, A, q_{P_3,13})^{n-2}(q_{P_3,13}, R, RA, q_{P_3,14})(q_{P_3,14}, x, x, q_{P_3,14})...$$

thus $P_3A^iSWLA^{2n}RSw_{rest} \Rightarrow_G P_3A^iSWALA^{n-2}RASw_{rest}$.

In case that $l > 0$, the first $i + 3$ computational steps of M are similar to the previous case. Computation then continues by:

$$...(q_{P_3,9}, A, W, q_{P_3,10})(q_{P_3,10}, A, A, q_{P_3,10})^{l-1}(q_{P_3,10}, L, A, q_{P_3,11})(q_{P_3,11}, A, AL, q_{P_3,12})$$
$$(q_{P_3,12}, A, \epsilon, q_{P_3,13})(q_{P_3,13}, A, A, q_{P_3,13})^{2n-2l-2}(q_{P_3,13}, R, RA, q_{P_3,14})(q_{P_3,14}, x, x, q_{P_3,14})...$$

thus the sentential form $P_3A^iSWA^{l+1}LA^{2n-2l-2}RA^{l+1}Sw_{rest}$ is derived.

For $l = n$ we have the sentential form $P_3A^iSWA^nLRA^nSw_{rest}$ on which the computation of M starts as in the previous case and then continues from the state $q_{P_3,11}$ as follows:

$$...(q_{P_3,11}, R, CS, q_{P_3,14})(q_{P_3,14}, x, x, q_{P_3,14})...$$

so the output is $P_3A^iSWA^nCSA^nSw_{rest}$. The fact that nonterminal $R$ follows immediately after $L$ in the sentential form indicates that they are located in the middle of the block $A^{2n}$. M replaces them by nonterminals $CS$, where $S$ separates nonterminals $A$ and $C$ is shifted to the left in the following steps in order to signalize that the middle of $A^{2n}$ was found.

As suggested, on the sentential form $P_3A^iSWA^lCA^{n-l}Sw_{rest}$ for any $i, l > 0$, $C$ is shifted to the left. After reading prefix $P_3A^iSWA^l$ 1-a-transducer M is in the state $q_{P_3,10}$ and word $P_3A^iSWA^{l-1}$ has been written to the output so far (similarly to the previous case). The rest of the computation looks as follows:

$$(q_{P_3,10}, C, CA, q_{P_3,14})(q_{P_3,14}, x, x, q_{P_3,14})...$$

so the sentential form $P_3A^iSWA^{l-1}CA^{n-l+1}Sw_{rest}$ is derived.

After certain number of such shifts $C$ appears next to the nonterminal $W$ in the sentential form. At this point, nonterminals $WC$ are deleted, the last block $A^n$ is transformed to $a_1^n$ and all terminal subwords are incremented in rlo. More formally, on the sentential form $P_3A^iSWCA^nSA^nSw_1S...Sw_lS$ for all $i > 0$ and $w_1, ..., w_l \in \Sigma^+$ M works as follows: on prefix $P_3A^iSW$ M moves to the state $q_{P_3,9}$ and writes $P_3A^iS$ so far as we have shown before. The computation then continues with arcs:

$$(q_{P_3,9}, C, \epsilon, q_{P_3,16})(q_{P_3,16}, A, A, q_{P_3,16})^n(q_{P_3,16}, S, S, q_{P_3,17})$$
$$(q_{P_3,17}, A, a_1, q_{P_3,17})^n(q_{P_3,17}, S, S, q_{P_3,18}),$$

so $A^nSa_1^nS$ is appended to the output. We shall show that from the state $q_{P_3,18}$ on the input $a_k^ra_jwS$ for any $r$, where $j < k$ and $w \in \Sigma^*$, M writes $a_1^ra_{j+1}wS$ and ends up again in the state $q_{P_3,18}$. In such case the computation of M is:

$$(q_{P_3,18}, a_k, a_1, q_{P_3,18})^r(q_{P_3,18}, a_j, a_{j+1}, q_{P_3,19})(q_{P_3,19}, a, a, q_{P_3,19})^{|w|}(q_{P_3,19}, S, S, q_{P_3,18})$$

where $a$ is arbitrary terminal from $\Sigma$. Thus the rest of the terminal words separated by $S$ is incremented in rlo so after this generative step the sentential form $P_3A^iSA^nSw_1'S...Sw_l'S$ is derived, where $w_s'$ is the next successor of $w_s$ in rlo for all $s \in \{1, ..., l\}$.

The last case we analyze is the input of a form $P_3SA^nSw_1S...Sw_{k^n}S$, where $w_1, ..., w_{k^n}$ are terminal words from $\Sigma^*$, on which M works as follows:

$$(q_0, P_3, \epsilon, q_{P_3,1})(q_{P_3,1}, S, \epsilon, q_{P_3,7})(q_{P_3,7}, A, P_4, q_{P_3,8})(q_{P_3,8}, A, \epsilon, q_{P_3,8})^{n-1}$$
$$(q_{P_3,8}, S, C\frac{B}{B}q_A, q_{P_3,5})(q_{P_3,5}, a, \frac{a}{a}, q_{P_3,5})...(q_{P_3,5}, S, \frac{B}{B}\$, q_{P_3,20})(q_{P_3,20}, a, \frac{B}{B}q_A\frac{a}{a}, q_{P_3,5})...$$

where $a$ stands for any terminal $a \in \Sigma$ and $q_A$ is the initial state of TM A. After such generative step the sentential form $P_4C\overline{w_1}\$...\overline{w_{n^k}}\$$ is derived.

From the above analysis we have that G works on $P_3A^{k^n}SA^nS$ as follows:

while $P_3$ is followed by $A$ in the sentential form, M enters the cycle:

$P_3A^jSA^nSw_{rest} \Rightarrow_G P_3A^{j-1}SWLA^{2n}RSw_{rest}$

While $L$ is not followed by $R$:

$P_3A^iSWA^lLA^{2n-2l}RA^lSw_{rest} \Rightarrow_G P_3A^iSWA^{l+1}LA^{2n-2l-2}RA^{l+1}Sw_{rest}$

$P_3A^iSWA^nLRA^nSw_{rest} \Rightarrow_G P_3A^iSWA^nCSA^nSw_{rest}$

While $W$ is followed by $A$ shift $C$ to the left:

$P_3A^iSWA^lCA^{n-l}Sw_{rest} \Rightarrow_G P_3A^iSWA^{l-1}CA^{n-l+1}Sw_{rest}$

$P_3A^iSWCA^nSA^nSw_1S...Sw_{k^n-i-1}S \Rightarrow_G P_3A^iSA^nSw_1S...Sw_{k_n-i}S$

where $w_s$ is the $s^{th}$ word in rlo from $\Sigma^n$ for all $s \in \{1, ..., k^n\}$. Note that this cycle is repeated $k^n$ times so after the cycle ends the derived sentential form contains all words from $\Sigma^n$ in rlo (separated by $S$). The last generative step of this phase is:

$$P_3SA^nSa_1^nS...Sa_k^nS \Rightarrow_G P_4C\overline{a_1^n}\$...\overline{a_k^n}\$$$

Phase 4: In this phase an arbitrary number of nonterminals $A$ is generated so the sentential form $P_5A^mC\overline{a_1^n}\$...\overline{a_k^n}\$$ for any $m$ is derived. For that purpose nondeterminism is used similarly to the construction from Lemma 5.8.

Phase 5: G simulates the work of G' on each of the derived words (its double track version) and subword $A^m$ is used to specify which word from the "accepted" ones will be generated in the terminal form.

Now let us analyze the work of M on the sentential form $P_5A^iC\frac{B}{B}zw_{first}\$w_{rest}$ for any $i > 0$, $z \in N \cup T$, $w_{first} \in ((N \cup T) - \{\$\})^*$, where $zw_{first}$ is double track word composed of input word on the first track and current configuration of simulated TM A on the second track. Word $w_{rest}$ contains the rest of such words delimited by \$ symbol. On the prefix $P_5A^iC\frac{B}{B}$ where $i > 0$ M works as follows:

$$(q_0, P_5, \epsilon, q_{P_5,1})(q_{P_5,1}, A, P_5, q_{P_5,2})(q_{P_5,2}, A, A, q_{P_5,2})^{i-1}(q_{P_5,2}, C, \epsilon, q_{P_5,3})(q_{P_5,3}, \frac{B}{B}, \epsilon, q_{P_5,4}).$$

Then the computation continues according to $z$, because M checks whether the first simulation reached accepting or rejecting configuration. For $z = q_{accept}$ we have:

$$(q_{P_5,4}, q_{accept}, C, q_{P_5,5})(q_{P_5,5}, y, \epsilon, q_{P_5,5})^{|w_{first}|}(q_{P_5,5}, \$, \epsilon, q_0')$$

and then M simulates one step of G' on the remaining subwords delimited by \$ which can be easily seen from the assumption that on endmarker symbol M' moves to $q_0'$. Thus $\frac{B}{B}zw_{first}\$$ is deleted from the sentential form and counter $A^i$ is decremented by one.

In case that $z = q_{reject}$ M continues with $(q_{P_5,4}, q_{reject}, AC, q_{P_5,5})$ and the rest is the same as in the previous case. Again, subword $\frac{B}{B}zw_{first}\$$ is deleted but this time counter is not decremented.

In the last case when $z \notin \{q_{accept}, q_{reject}\}$ M uses the arc $(q_{P_5,4}, z, ACv, q)$ such that the arc $(\left[\frac{B}{B}\right], z, v, q) \in H'$ and simulates one generative step of G' on each subword including $zw_{first}\$$.

On similar sentential form but without nonterminals $A$ the computation of M starts as follows:

$$(q_0, P_5, \epsilon, q_{P_5,1})(q_{P_5,1}, C, \epsilon, q_{P_5,6})(q_{P_5,6}, \frac{B}{B}, \epsilon, q_{P_5,7})$$

and then, again, computation differs according to $z$. On $z \notin \{q_{accept}, q_{reject}\}$ M continues with $(q_{P_5,7}, z, P_5Cv, q)$ such that $(\left[\frac{B}{B}\right], z, v, q) \in H'$ so one step of G' is simulated on the rest similarly to the previou case.

In case $z = q_{reject}$ the following arc is $(q_{P_5,7}, q_{reject}, P_5C, q_{P_5,5})$ and then $w_{first}$ is deleted and one step of G' is simulated on the rest of the sentential form in aforementioned way.

The last case is $z = q_{accept}$ on which the computation continues by

$$(q_{P_5,7}, q_{accept}, \epsilon, q_{P_5,8})(q_{P_5,8}, \frac{a}{x}, a, q_{P_5,8})...(q_{P_5,8}, \frac{B}{x}, \epsilon, q_{P_5,8})...(q_{P_5,8}, \$, \epsilon, q_{P_5,9})(q_{P_5,9}, x, \epsilon, q_{P_5,9})...$$

where $a$ represents any terminal and $x$ any symbol from $N \cup T$. Thus terminal sentential form of the word from the first track of $w_{first}$ is derived.

From the above analysis we can see that in this phase G from the sentential form

$$P_5 A^m C \overline{a_0^n}\$...\overline{a_k^n}\$$$

derives the $m+1^{th}$ subword from the first track which is accepted by TM A.

Now that we made the analysis of all phases we shall show the correctness of the construction. To prove the inclusion $L \subseteq L(G)$ we find the derivation of arbitrary $w \in L$ in G. Let $n = |w|$ and let $m = \#v \mid v \in L, |v| = n, v \ precedes \ w \ in \ rlo$. The derivation of $w$ in G looks as follows:

$$\begin{aligned}
\sigma &\Rightarrow_G^* P_2 A^n P_3 ASA^n S && (phase\ 1)\\
&\Rightarrow_G^* P_3 A^{k^n} SA^n S && (phase\ 2)\\
&\Rightarrow_G^* P_4 C \overline{a_1^n}\$...\overline{a_k^n}\$ && (phase\ 3)\\
&\Rightarrow_G^* P_5 A^m C \overline{a_1^n}\$...\overline{a_k^n}\$ && (phase\ 4)\\
&\Rightarrow_G^* w && (phase\ 5)
\end{aligned}$$

Now we prove the opposite inclusion $L \supseteq L(G)$: for any word $w$ generated by G it holds that G' derives from $\frac{B}{B} q_A \frac{w}{w} \frac{B}{B}\$$ the sentential form starting with $\frac{B}{B} q_{accept}$ thus $w \in L(A)$ and G will halt on any sentential form starting with terminal. Both inclusions hold so $L = L(G)$.

Let us count the number of nondeterministic decisions made during the derivation of $w \in L(G), |w| = n$. In the phase 1 G can derive sentential form $P_2 A^n P_3 ASA^n S$ using $log(n)$ decisions (similarly to Lemma 5.8). In the phase 4 this number is $log(m)$ (for $m$ generated nonterminals $A$) as well so the last question is what is the biggest meaningful $m$? From the analysis of G we can see that in phase 5 the number of nonterminals $A$ in the sentential form is decreased when the simulation of the first subword in the sentential form reaches the acceptance - prefix $P_5 A^i C \frac{B}{B} q_{accept}$. Thus the maximal meaningful number of nonterminals $A$ generated in phase 4 is the number of distinct words of the given length that are accepted by the Turing machine $A$. Thus we have that $lengthDec(G, n) = log(n) + log(|W_n|)$ where $W_n = \{w | w \in L, |w| = n\}$ and the theorem follows.

□

# Chapter 6

# Conclusion

A property of languages generated by dgs related to prefixes was proven in Chapter 3 which brings more light to the reason why are deterministic generative systems so weak and why adding the endmarker helps to improve their generative power.

In Chapter 4 we presented a construction of a \$dgs G for arbitrary Turing machine A such that $L(G) = L(A)$. The main idea of this construction was to generate the words from $L(A)$ in order given by the number of computational steps of $A$ on which they are accepted. This enabled us to simulate only limited number of steps of $A$ and to avoid the non halting simulations.

We introduced two computational measures of nondeterminism in generative systems based on the lenght of the derivation and on the length of the derived word in Chapter 5. We showed that between two nondeterministic generative steps an arbitrary number of deterministic steps can be inserted. It also turns out that nondeterminism is needed mainly to derive arbitrary word from $\Sigma^*$ and verification whether this word is accepted by a given Turing machine can be done deterministically.

Some questions from [1] about deterministic g-systems remain open despite our effort to solve them. Namely, it remains unclear whether the family of languages $\mathcal{L}_{DG}$ is closed under $\cap R$ and whether $\mathcal{L}_{DG} - \mathcal{L}_{CS} = \emptyset$.

# Bibliography


[1] Martin Králik. Deterministic generative systems. Master's thesis. Comenius University, Bratislava, Faculty of Mathematics, Physics and Informatics. 2002.

[2] Dušan Krcho. Non-determinism in Generative Systems. Master's thesis. Comenius University, Bratislava, Faculty of Mathematics, Physics and Informatics. 2002.

[3] Branislav Rovan. A framework for studying grammars. In *International Symposium on Mathematical Foundations of Computer Science*, LNCS 118, pages 473–482. Springer, 1981.