Distance Desert Automata and Star Height Substitutions

Der Fakultät Mathematik und Informatik

der Universität Leipzig

eingereichte

Habilitationsschrift

zur Erlangung des akademischen Grades

doctor rerum naturalium habilitatus

(Dr. rer. nat. habil.)

vorgelegt

von

Dr. rer. nat. Daniel Kirsten

geboren am 23.4.1972 in Merseburg/Saale.

Die Annahme der Habilitationsschrift haben empfohlen:

1. Prof. Dr. rer. nat. habil. Manfred Droste, Universität Leipzig
2. Prof. Jean-Éric Pin, CNRS et Université Paris VII
3. Prof. Dr. rer. nat. habil. Wolfgang Thomas, RWTH Aachen

Beschluss über die Verleihung des akademischen Grades vom 27. Februar 2006

Abstract

We introduce the notion of nested distance desert automata as a joint generalization and further development of distance automata and desert automata. We show that limitedness of nested distance desert automata is PSPACE-complete.

As an application, we show that it is decidable in $2^{2^{\mathcal{O}(n^2)}}$ space whether the language accepted by an n-state non-deterministic automaton is of a star height less than a given integer h (concerning rational expressions with union, concatenation and iteration). We also show some decidability results for some substitution problems for recognizable languages.

Acknowledgements

First I would like to thank my wife Ines for all her love and care during the recent years.

I thank my supervisors Jean-Éric Pin and Manfred Droste for their support. I thank all my colleagues and friends in Dresden, Leipzig, and Paris for fruitful discussions.

I thank the members of the habilitation commission Gerhard Heyer, Gerhard Brewka, Heinrich Herre, Dietrich Kuske, and Gerik Scheuermann for their regular and careful execution of my habilitation procedure. I thank the referees of this thesis Manfred Droste, Jean-Éric Pin, and Wolfgang Thomas for their careful work.

I thank Christian Choffrut and his wife Conchita for their support during my stay in Paris in 2003.

I thank the Freistaat Sachsen for providing my position at Dresden University of Technology and financing various journeys. I greatly acknowledge the financial support by the German Research Foundation (Deutsche Forschungsgemeinschaft) which funded a 12 month research grant to work at LIAFA in Paris. In this period, I developed the key ideas and main results of this thesis. I thank the CNRS and the NATO Security Through Science Programme for supporting journeys.

This thesis was submitted on January 24, 2005.

Contents

Part I
Complete Version

Part I contains the complete version of the Habilitationsschrift (postdoctoral thesis) written in english language.

1 Introduction

The *star height problem* was raised by L.C. Eggan in 1963 [7]: Is there an algorithm which computes the star height of recognizable languages? Like L.C. Eggan, we consider star height concerning rational expressions with union, concatenation, and iteration in contrast to extended star height which also allows intersection and complement. For several years, in particular after R. McNaughton refuted some promising ideas in 1967 [39], the star height problem was considered as the most difficult problem in the theory of recognizable languages, and it took 25 years until K. Hashiguchi showed the existence of such an algorithm which is one of the most important results in the theory of recognizable languages [14]. However, [14] is very difficult to read, e.g., J.-É. Pin commented "Hashiguchi's solution for arbitrary star height relies on a complicated induction, which makes the proof very difficult to follow." [48]. The entire proof stretches over [11, 12, 13, 14], and I. Simon mentioned that it "takes more than a hundred pages of very heavy combinatorial reasoning" to present K. Hashiguchi's solution in a self contained fashion [51]. D. Perrin wrote "the proof is very difficult to understand and a lot remains to be done to make it a tutorial presentation" [43].

K. Hashiguchi's solution to the star height problem yields an algorithm of non-elementary complexity, and it remains open to deduce any upper complexity bound from K. Hashiguchi's approach (cf. [36, Annexe B]).

Motivated by his research on the star height problem, K. Hashiguchi introduced the notion of distance automata in 1982 [11, 12]. *Distance automata* are nondeterministic finite automata with a set of marked transitions. The weight of a path is defined as the number of marked transitions in the path. The weight of a word is the minimum of the weights of all successful paths of the word. Thus, distance automata compute mappings from the free monoid to the positive integers. K. Hashiguchi showed that it is decidable whether a distance automaton is limited, i.e., whether the range of the computed mapping is finite [11].

Distance automata and the more general weighted automata over the tropical semiring became a fruitful concept in theoretical computer science with many applications beyond their impact for the decidability of the star height hierarchy [14], e.g., they have been of crucial importance in the research on the star problem in trace monoids [26, 40], but they are also of interest in industrial applications as speech recognition [41], database theory [9], and image compression [5, 21]. Consequently, distance automata and related concepts have been studied by many researchers beside K. Hashiguchi, e.g., [15, 17, 27, 31, 35, 50, 51, 53, 54, 55].

S. Bala and the author introduced independently the notion of desert automata in [1, 2, 22, 25]. *Desert automata* are nondeterministic finite automata with a set of marked transitions. The weight of a path is defined as the length of a longest factor which does not contain a marked transition. The weight of a word is the minimum of the weights of all successful paths of the word. S. Bala and the author showed that limitedness of desert automata is decidable [1, 2, 22, 25]. As an application, S. Bala and the author solved the so-called finite substitution problem which was open for more than 10 years: given recognizable languages K and L, it is decidable whether there exists a finite substitution σ such that $\sigma(K) = L$ [1, 2, 22, 25].

Here, we introduce a joint generalization of distance automata and desert automata, the *nested distance desert automata*. By a generalization and further development of approaches from [15, 22, 23, 25, 29, 30, 34, 50, 51, 53], we develop two characterizations of unlimited nested distance desert automata and show that their limitedness problem is PSPACE-complete. To achieve the decidability of limitedness in polynomial space we positively answer a question from H. Leung's

PhD thesis [29] from 1987.

As an application of nested distance desert automata, we give a new proof and the first upper complexity bound for the star height problem: given an integer h and an n-state nondeterministic automaton $\mathcal{A}$, it is decidable in $2^{2^{\mathcal{O}(n^2)}}$ space whether the star height of the language of $\mathcal{A}$ is less than h. The complexity bound does not depend on h because the star height of the language of an n-state nondeterministic automaton cannot exceed n.

Our approach to the star height problem allows to develop several results on substitution problems. A substitution is of star height h if it maps letters to recognizable languages of a star height of at most h. We prove that given some integer h and recognizable languages K and L, it is decidable in double exponential space whether there is a substitution σ of star height of at most h such that $\sigma(K) = L$. This result contains the decidability of the finite substitution problem, the so-called recognizable substitution problem as well as the star height problem.

This thesis is organized as follows. In Section 2, we state preliminary notions and introduce nested distance desert automata. We present our main results in Section 2.4 and 2.5. In Section 3, we get familiar with some algebraic and technical foundations. In particular, we recall classic notions from ideal theory and the notion of a consistent mapping and we develop a finite semiring to describe nested distance desert automata in an algebraic fashion.

In Section 4, we develop two characterizations of unlimited nested distance desert automata which generalize classic results for distance automata due to K. Hashiguchi, H. Leung, and I. Simon. One of these characterization utilizes K. Hashiguchi's so-called $\sharp$-expressions [15]. The other characterization relies on a solution of a so-called Burnside type problem which is similar to H. Leung's and I. Simon's approaches to the limitedness of distance automata. It gives immediately the decidability of limitedness of nested distance desert automata. To show the decidability of limitedness in PSPACE, we develop some more ideas in Section 5.

In Section 6, we reduce the star height problem to limitedness of nested distance desert automata. In Section 7, we consider star height substitutions. In Section 8, we discuss our approach and point out some open questions. Sections 6 and 7 can be read independently of Sections 3, 4, 5.

If the reader is just interested in *how* to decide limitedness of distance desert automata, he should read Section 2, Section 3.3, and Section 4.1 to get familiar with a decidable characterization of umlimited distance desert automata. To understand the correctness of this characterization, he should read Sections 3 and 4. To understand how this decidable characterization yields a PSPACE-algorithm, he has to read additionally Section 5.

If the reader is just interested in the star height problem, he may skip Sections 3, 4, and 5 and read Section 6. Section 7 relies on the constructions in Section 6.

This thesis is almost self-contained. The reader should be familiar with some basic notions on recognizable languages as Kleene's theorem and the syntactic monoid. We need some very basic notions from complexity theory, e.g., PSPACE-completeness. However, these notions are required just to understand the complexity results. All the results in this thesis including the *upper* bound on the complexity of the star height problem are developed from scratch and it is possible to understand all the constructions without consulting earlier papers on the star height problem up to one exception: to show the *lower* bound on the complexity of the star height problem, we use the strictness of the star height hierarchy due to F. Dejean and M. Schützenberger [6] and we borrow a certain encoding from K. Hashiguchi and N. Honda [19].

2 Overview

2.1 Preliminaries

Let $\mathbb{N} = \{0, 1, \dots\}$. For finite sets M, we denote by $|M|$ the number of elements of M. If p belongs to some set M, then we denote by p both the element p and the singleton set consisting of p. For sets M, we denote by $\mathcal{P}(M)$ the power set of M, and we denote by $\mathcal{P}_{ne}(M)$ the set of all non-empty subsets of M. We denote the union of disjoint sets by $\uplus$.

A *semigroup* $(S, \cdot)$ consists of a set S and a binary associative operation $\cdot$. Usually, we denote $(S, \cdot)$ for short by S, and we denote the operation $\cdot$ by juxtaposition.

Let S be a semigroup. We call S *commutative*, if $ab = ba$ for every $a, b \in S$. We call an element $1 \in S$ an *identity*, if we have for every $a \in S$, $1a = a1 = a$. If S has an identity, then we call S a *monoid*. We call an element $0 \in S$ a *zero*, if we have for every $a \in S$, $a0 = 0a = 0$. There are at most one identity and at most one zero in a semigroup. We extend the operation of S to subsets of S in the usual way.

For subsets $T \subseteq S$, we call the closure of T under the operation of S the *subsemigroup generated by* T and denote it by $\langle T \rangle$. If $\langle T \rangle = T$, then we call T a *subsemigroup* of S.

Let $\leq$ be a binary relation over some semigroup S. We call $\leq$ *left stable* (resp. *right stable*) if for every $a, b, c \in S$ with $a \leq b$ we have $ca \leq cb$ (resp. $ac \leq bc$). We call $\leq$ *stable* if it is both left stable and right stable.

A *semiring* $(K, +, \cdot)$ consists of a set K and two binary operations $+$ and $\cdot$ whereas $(K, +)$ is a commutative monoid with an identity 0, $(K, \cdot)$ is a semigroup with zero 0, and the distributivity laws hold, i.e., for every $a, b, c \in K$, we have $a(b + c) = ab + ac$ and $(b + c)a = ba + ca$. Note that we do not require that a semiring has an identity for $\cdot$. A semiring $(K, +, \cdot)$ id called *commutative*, if $(S, \cdot)$ is a commutative semigroup.

During the main part of this thesis, we fix some $n \geq 1$ which is used as the dimension of matrices. Whenever we do not explicitly state the range of a variable, then we assume that it ranges over the set $\{1, \dots, n\}$. For example, phrases like "for every i, j" or "there is some l, such that" are understood as "for every $i, j \in \{1, \dots, n\}$" resp. "there is some $l \in \{1, \dots, n\}$, such that".

If $(K, +, \cdot)$ is a semiring, then we denote by $K_{n \times n}$ the semiring of all $n \times n$-matrices over K equipped with matrix multiplication (defined by $\cdot$ and $+$ as usual) and componentwise operation $+$.

2.2 Words, Classic Automata, ♯-Expressions, and Substitutions

We recall some terminology in formal language theory.

Let Σ be a finite set of symbols. We denote by Σ^* the free monoid over Σ, i.e., Σ^* consists of all *words* over Σ with concatenation as operation. We denote the empty word by ε. We denote by Σ^+ the free semigroup over Σ, i.e., $\Sigma^+ := \Sigma^* \setminus \varepsilon$. For every $w \in \Sigma^*$, we denote by $|w|$ the length of w. We call subsets of Σ^* *languages*. We call a word u a *factor* of a word w if $w \in \Sigma^* u \Sigma^*$. Let $u, v \in \Sigma^*$. Every factor of uv is the concatenation of a factor of u and a factor of v. For instance, if $u = aaab$ and $v = bb$, then the factor aa of $uv = aaabbb$ is the concatenation of aa and ε which are factors of u resp. v.

For $L \subseteq \Sigma^*$, we define $L^* := L^0 \cup L^1 \cup \dots = \cup_{i \in \mathbb{N}} L^i$ and $L^+ := L^1 \cup L^2 \cup \dots = \cup_{i \geq 1} L^i$. Note that regardless of L, we have $L^0 = \{\varepsilon\}$. We call L^* the *iteration of* L.

Note that M^* is defined in two ways, depending on whether M is a set of symbols or M is a language. However, we will use the notation M^* in a way that no confusion arises.

A (nondeterministic) *automaton* is a tuple $\mathcal{A} = [Q, E, I, F]$ where

1. Q is a finite set of *states*,
2. $E \subseteq Q \times \Sigma \times Q$ is a set of *transitions*, and
3. $I \subseteq Q$, $F \subseteq Q$ are sets called *initial* resp. *accepting states*.

Let $k \geq 1$. A path π in $\mathcal{A}$ of length k is a sequence $(q_0, a_1, q_1)\,(q_1, a_2, q_2)\,\ldots\,(q_{k-1}, a_k, q_k)$ of transitions in E. We say that π starts at q_0 and ends at q_k. We call the word $a_1 \ldots a_k$ the *label of* π. We denote $|\pi| := k$. As usual, we assume for every $q \in Q$ a path which starts and ends at q and is labeled with ε.

We call π *successful* if $q_0 \in I$ and $q_n \in F$. For every $0 \leq i \leq j \leq k$, we denote $\pi(i,j) := (q_i, a_i, q_{i+1})\,\ldots\,(q_{j-1}, a_{i-1}, q_j)$ and call $\pi(i,j)$ a *factor of* π. For every $p, q \in Q$ and every $w \in \Sigma^*$, we denote by $p \overset{w}{\rightsquigarrow} q$ the set of all paths with the label w which start at p and end at q. In the same way, we denote for every $P, R \subseteq Q$ and $w \in \Sigma^*$ by $P \overset{w}{\rightsquigarrow} R$ the set of all paths with the label w which start at some state in P and end at some state in R.

We denote the *language of* $\mathcal{A}$ by $L(\mathcal{A})$ and define it as the set of all words in Σ^* which are labels of successful paths. We call some language $L \subseteq \Sigma^*$ *recognizable*, if L is the language of some automaton. See, e.g, [3, 8, 56] for a survey on recognizable languages.

For every automaton $\mathcal{A} = [Q, E, I, F]$ such that $\varepsilon \notin L(\mathcal{A})$, one can easily construct an automaton $\mathcal{A}' = \big[Q', E', \{q_I\}, \{q_F\}\big]$ such that $Q' = Q \cup \{q_I, q_F\}$, $E' \subseteq (Q \setminus \{q_F\}) \times \Sigma \times (Q \setminus \{q_I\})$, and $L(\mathcal{A}) = L(\mathcal{A}')$.

Let $L \subseteq \Sigma^*$ be a language and M be a monoid. We say that M recognizes L if there is some homomorphism $\eta : \Sigma^* \to M$ such that $L = \eta^{-1}(\eta(L))$. It is well-known that a language L is recognizable iff L is recognized by some finite monoid. The smallest monoid which recognizes L is called the *syntactic monoid of* L. If L is effectively given, e.g., by some nondeterministic automaton $[Q, E, I, F]$, then one can effectively construct the syntactic monoid M (and a surjective homomorphism η such that $\eta^{-1}(\eta(L)) = L$), and we have $|M| \leq 2^{|Q|^2}$.

The notion of a $\sharp$-expression was introduced by K. Hashiguchi in 1990 [15]. Intuitively, $\sharp$-expressions provide a nested pumping technique. Every letter $a \in \Sigma$ is a *$\sharp$-expression*. For $\sharp$-expressions r and s, the expressions rs and $r^\sharp$ are *$\sharp$-expressions*.

Let r be a $\sharp$-expression. For every $k \geq 1$, r defines a word $r(k)$. If r is just a letter, then $r(k) := r$. For $\sharp$-expressions r and s, we set $rs(k) := r(k) \cdot s(k)$. For every $\sharp$-expression r, $r^\sharp(k)$ yields the k-th power of $r(k)$, i.e., we define $r^\sharp(k) := \big(r(k)\big)^k$.

The $\sharp$-height of $\sharp$-expressions is defined inductively. Letters are of $\sharp$-height 0, the $\sharp$-height of rs is the maximum of the $\sharp$-heights of r and s, and the $\sharp$-height of $r^\sharp$ is the $\sharp$-height of r plus 1.

Let X be an alphabet disjoint from Σ. We call every mapping $\sigma : X \to \mathcal{P}(\Sigma^*)$ a *substitution*. Let σ be a substitution. By setting $\sigma(a) := a$ for every $a \in \Sigma$, σ generalizes to a unique homomorphism $\sigma : \big(\mathcal{P}(\Sigma \cup X)^*, \cup, \cdot\big) \to \big(\mathcal{P}(\Sigma^*), \cup, \cdot\big)$.

A substitution σ is called *non-erasing* (resp. *non-empty* resp. *finite* resp. *recognizable*) if for every $x \in X$, we have $\varepsilon \notin \sigma(x)$ (resp. $\sigma(x) \neq \emptyset$ resp. $\sigma(x)$ is finite resp. $\sigma(x)$ is recognizable).

2.3 Nested Distance Desert Automata

Let $h \geq 0$ be arbitrary and $V := \{\angle_0, \curlyvee_0, \angle_1, \curlyvee_1, \ldots, \curlyvee_{h-1}, \angle_h\}$. We define a mapping $\Delta : V^* \to \mathbb{N}$ in a tricky way. Before we define Δ formally, we give an intuitive explanation. We regard the numbers $0, \ldots, h$ as colors. For every $0 \leq g \leq h$, there are coins of color g which are called g-coins. We have some bag to carry coins. The bag has exactly $h + 1$ partitions which are colored like the coins. For every $0 \leq g \leq h$, we can store g-coins in partition g, but we cannot store g-coins in any other partition. The *size of the bag* is an integer d whereas we can carry at most d 0-coins, d 1-coins, ..., and d h-coins at the same time. Hence, we can carry at most $d(h+1)$ coins at the same time, but we cannot carry more than d coins of one and the same color at the same time.

Imagine that we plan to walk along some word[1] $\pi \in V^*$, and we have a bag of size $d \in \mathbb{N}$. Initially, the bag is completely filled, i.e., there are d 0-coins, d 1-coins, ..., and d h-coins in the bag. Let $0 \leq g \leq h$ be arbitrary. If we walk along the letter $\angle_g$, then we have to pay a g-coin but we can obtain coins which are colored by a color less than g, i.e., we can fill up our bag with 0-coins, 1-coins, ..., $(g-1)$-coins. If we do not carry a g-coin in our bag, then we cannot walk along the letter $\angle_g$. We pronounce the letter $\angle_g$ as "péage g". If we walk along $\curlyvee_g$, then we need not to pay any coin but we can fill up our bag with 0-coins, 1-coins, ..., g-coins. We pronounce $\curlyvee_g$ as "water g". This notion arose from earlier variants of these automata in which $\curlyvee$ was considered as a source of water.

Whenever we can obtain g-coins (at $\curlyvee_g, \angle_{g+1}, \ldots, \curlyvee_{h-1}, \angle_h$), then we can also obtain 0, ..., $(g-1)$-coins. However, at $\curlyvee_{g-1}$ and $\angle_g$, we can obtain $0, \ldots, (g-1)$-coins but we cannot obtain g-coins. Thus, g-coins are considered as more valuable than $0, \ldots, (g-1)$-coins.

It depends on the size of the bag (and of course on the word) whether we can walk along the entire word. We imagine $\Delta(\pi)$ as the least integer d such that we can walk along the word π with a bag of size d.

We define Δ formally. For every $0 \leq g \leq h$, we consider every factor π' of π in which we cannot obtain g-coins. More precisely, we consider factors π' of π with $\pi' \in \{\angle_0, \curlyvee_0, \ldots, \angle_g\}^*$ and count the number of occurrences of $\angle_g$. This is the number of g-coins which we need to walk along π'.

For $0 \leq g \leq h$ and $\pi \in V^*$, let $|\pi|_g$ be the number of occurrences of the letter $\angle_g$ in π. Let

1. $\Delta_g(\pi) := \max_{\substack{\pi' \in \{\angle_0, \curlyvee_0, \ldots, \angle_g\}^* \\ \pi' \text{ is a factor of } \pi}} |\pi'|_g$ and

2. $\Delta(\pi) := \max_{0 \leq g \leq h} \Delta_g(\pi)$.

It is easy to see that $\Delta(\pi) \leq |\pi|$.

An *h-nested distance desert automaton* is a tuple $\mathcal{A} = [Q, E, I, F, \theta]$ where $[Q, E, I, F]$ is an automaton and $\theta : E \to V$.

Let $\mathcal{A} = [Q, E, I, F, \theta]$ be an h-nested distance desert automaton. The notions of a path, a successful path, the language of $\mathcal{A}$, ... are understood w.r.t. $[Q, E, I, F]$. For every transition $e \in E$, we say that e is *marked by* $\theta(e)$. We extend θ to a homomorphism $\theta : E^* \to V^*$. We define the semantics of $\mathcal{A}$. For every $w \in \Sigma^*$, let

$$\Delta_{\mathcal{A}}(w) := \min_{\pi \in I \overset{w}{\rightsquigarrow} F} \Delta(\theta(\pi)).$$

In particular, $\Delta_{\mathcal{A}}(w) = \infty$ iff $w \notin L(\mathcal{A})$. Hence, $\Delta_{\mathcal{A}}$ is a mapping $\Delta_{\mathcal{A}} : \Sigma^* \to \mathbb{N} \cup \{\infty\}$.

If there is a bound $d \in \mathbb{N}$ such that $\Delta_{\mathcal{A}}(w) \leq d$ for every $w \in L(\mathcal{A})$, then we say that $\mathcal{A}$ is *limited by d* or for short $\mathcal{A}$ is *limited*. Otherwise, we call $\mathcal{A}$ *unlimited*.

Clearly, h-nested distance desert automata are a particular case of $(h+1)$-nested distance desert automata.

For every 0-nested distance desert automaton $\mathcal{A}$, we have $\Delta_{\mathcal{A}}(w) = |w|$ for every $w \in L(\mathcal{A})$. Hence, 0-nested distance desert automaton $\mathcal{A}$ is limited iff $L(\mathcal{A})$ is finite.

[1] Note that we utilize the letter π both to denote words over V but also to denote paths in automata.

The proper subclass of 1-nested distance desert automata for which $\theta : E \to \{\curlyvee_0, \angle_1\}$ are exactly K. Hashiguchi's distance automata [11]. If we consider the proper subclass of 1-nested distance desert automata with the restriction $\theta : E \to \{\angle_0, \curlyvee_0\}$, then we recover the definition of desert automata due to S. Bala and the author [1, 2, 22, 25].

2.4 Main Results on Nested Distance Desert Automata

Section 2.4 and 2.5 give an overview of the main results of this thesis.

One main result is a two-fold characterization of unlimited nested distance desert automata shown in Theorem 2.1, below. It generalizes results and ideas on distance and desert automata by K. Hashiguchi, H. Leung, I. Simon, and the author [15, 22, 23, 25, 29, 30, 34, 50, 51, 53]. Our first characterization is algebraic. It generalizes corresponding characterizations of unlimited distance automata due to H. Leung and I. Simon [29, 30, 34, 53] and unlimited desert automata due to the author [22, 25].

Our second characterization generalizes another well-known characterization of unlimited distance automata [15, 31, 34, 53] in terms of $\sharp$-expressions.

Theorem 2.1. *Let $h \in \mathbb{N}$. Let $\mathcal{A} = [Q, E, I, F, \theta]$ be a h-nested distance desert automaton. The following assertions are equivalent:*

1. *$\mathcal{A}$ is unlimited.*
2. *Let $T := \Psi(\Sigma)$. There is a matrix $a \in \langle T\rangle^\sharp$ such that $I \cdot a \cdot F = \omega$.*
3. *There is a $\sharp$-expression r of a $\sharp$-height of at most $(h+1)|Q|$ such that for every $k \geq 1$, we have $r(k) \in L(\mathcal{A})$, and for increasing integers k, the weight $\Delta_{\mathcal{A}}\big(r(k)\big)$ is unbounded.*

The algebraic concepts involved in assertion (2) in Theorem 2.1 will be explained in Section 3.3 and 4.1. At this point, it is not necessary to understand assertion (2).

Note that (3)⇒(1) in Theorem 2.1 is obvious. We prove (2)⇒(3) in Section 4.2 up to the bound on the $\sharp$-height of r which is considered in Section 5.3. The most difficult part in the proof of Theorem 2.1 is to show (1)⇒(2). It leads to an intriguing Burnside type problem and is shown in Section 4.6.

From Theorem 2.1, we derive the following result:

Theorem 2.2. *For $h \geq 1$, limitedness of h-nested distance desert automata is PSPACE-complete.*

Theorem 2.2 generalizes recent results due to H. Leung and V. Podolskiy [35] resp. S. Bala and the author [1, 2, 22, 25] for PSPACE-completeness for limitedness of distance resp. desert automata. However, the proof of the decidability of limitedness of nested distance desert automata in PSPACE is not a generalization of these two particular cases, it is an new approach which is based on an analysis of the structure of the semigroup $\langle T\rangle^\sharp$ in assertion (2) of Theorem 2.1. In particular, we will positively answer a question from H. Leung's PhD thesis from 1987 [29] (see Corollary 5.6(2)).

Theorem 2.2 will be proved in Section 5. We show a nondeterministic PSPACE-algorithm which decides limitedness of nested distance desert automata in Section 5.4. PSPACE-hardness for $h \geq 1$ in Theorem 2.2 follows immediately from PSPACE-hardness of limitedness of distance automata [29, 30] and of desert automata [22, 25]. However, we use H. Leung's idea [29, 30] show PSPACE-hardness of limitedness of some more particular cases of nested distance desert automata in Section 5.5.

Limitedness of 0-nested distance desert automata is essentially the question whether $L(\mathcal{A})$ is finite which is decidable in polynomial time.

The equivalence problem for distance automata is undecidable [27], and hence, the equivalence problem for h-nested distance desert automata is undecidable for $h \geq 1$. The equivalence problem for 0-nested distance desert is essentially the question whether $L(\mathcal{A}) = L(\mathcal{A}')$ which is PSPACE-complete. The equivalence problem for desert automata is open [22, 25].

2.5 Main Results on the Star Height Problem

We denote the *star height* of a rational expression r by $\mathsf{sh}(r)$. Every word in $w \in \Sigma^*$ is a rational expression of star height 0, i.e., $\mathsf{sh}(w) := 0$. Moreover, $\emptyset$ is a rational expression of *star height* 0. If r and s are rational expressions over Σ^*, then rs and $r \cup s$ are rational expressions of star height $\max\{\mathsf{sh}(r), \mathsf{sh}(s)\}$, but r^* is of star height $\mathsf{sh}(r) + 1$.

For every $k \in \mathbb{N}$, we define $\mathcal{L}_k := \{ L(r) \,|\, \mathsf{sh}(r) \leq k \}$. The class $\mathcal{L}_0$ consists of all finite languages. We denote the star height of a recognizable language L by $\mathsf{sh}(L)$ and define it as the least $k \in \mathbb{N}$ for which $L \in \mathcal{L}_k$. Already in 1963, L.C. Eggan showed $\mathcal{L}_k \subsetneq \mathcal{L}_{k+1}$ for every $k \in \mathbb{N}$, but he used an alphabet with $2^{k+1} - 1$ letters to construct a language in $\mathcal{L}_{k+1} \setminus \mathcal{L}_k$ [7]. In the same paper, he raised the *star height problem*:

1. Is the inclusion $\mathcal{L}_k \subseteq \mathcal{L}_{k+1}$ strict for every $k \in \mathbb{N}$ for $\Sigma = \{a, b\}$?
2. Is there an algorithm which computes the star height of recognizable languages?

During the recent 40 years, many papers have dealt with the star height problem. For a more detailed historical overview, the reader is referred to [45, 46, 51].

In 1966, F. Dejean and M. Schützenberger solved the first question by showing $\mathcal{L}_k \subsetneq \mathcal{L}_{k+1}$ for every $k \in \mathbb{N}$ for the alphabet $\Sigma = \{a, b\}$ [6]. In 1982, K. Hashiguchi showed that it is decidable whether a given recognizable language is of star height one [12, 13], and in 1988, he showed that the star height of recognizable languages is effectively computable [14]. Although this is a positive answer to the star height problem, there is still research for a better comprehension [36, 37, 38, 42]. This research aims at a deeper understanding of the star height problem for particular classes of recognizable languages, e.g., reversible languages.

Here, we give a new solution and the first upper bound for the complexity of the star height problem by a reduction to limitedness of nested distance desert automata.

Theorem 2.3. *Let $h \in \mathbb{N}$ and L be the language accepted by an n-state nondeterministic automaton. It is decidable in $2^{2^{\mathcal{O}(n^2)}}$ space whether L is of star height h.*

We prove Theorem 2.3 in Section 6 by a reduction of the star height h problem to limitedness of h-nested distance desert automata. Note that this reduction is immediate for $h = 0$, because a language L is of star height 0 iff L is finite, and the finiteness problem of a language is exactly the limitedness of 0-nested distance desert automata.

The complexity in Theorem 2.3 does not depend on h since from any proof of Kleene's theorem it follows that the star height of the language of an n-state nondeterministic automaton is at most n [56], and hence, the algorithm can avoid any computation if $h \geq n$ in Theorem 2.3.

We show a lower complexity bound for the star height problem in Section 6.5.

Theorem 2.4. *Let $h \geq 1$. To decide whether for a nondeterministic automaton $\mathcal{A}$ over a two letter alphabet, we have $\mathsf{sh}(L(\mathcal{A})) \leq h$ is PSPACE-hard.*

One can improve the space complexity in Theorem 2.3 to $2^{2^{\mathcal{O}(n)}}$ [24]. This is possible since the approach in [24] is based on a deterministic automaton for L while our approach in Section 6 of this thesis is based on the syntactic monoid of L. The advantage of the approach in this thesis is that we can develop some results on substitution problems.

The *star height* of a recognizable substitution $\sigma : X \to \mathcal{P}(\Sigma^*)$ is denoted by $\mathsf{sh}(\sigma)$ and defined as the maximal star height of the languages $\sigma(x)$ over $x \in X$. We show in Section 7:

Theorem 2.5.

1. *Given $h \geq 0$, languages $K \subseteq (\Sigma \cup X)^*$ and $L \subseteq \Sigma^*$ which are recognized by nondeterministic automata $[Q_K, E_K, I_K, F_K]$ and $[Q_L, E_L, I_L, F_L]$, it is decidable whether there exists a recognizable substitution $\sigma : X \to \mathcal{P}(\Sigma^*)$ satisfying $\sigma(K) = L$ and $\mathsf{sh}(\sigma) \leq h$. There is an algorithm to decide the existence of σ whose space complexity is polynomial in $|E_K| \cdot 2^{2^{\mathcal{O}(|Q_L|^2)}}$.*
2. *Given h, K, L as in assertion (1), it is decidable in the same complexity as in (1) whether there exists a non-erasing (resp. non-empty resp. non-erasing and non-empty) substitution $\sigma : X \to \mathcal{P}(\Sigma^*)$ satisfying $\sigma(K) = L$ and $\mathsf{sh}(\sigma) \leq h$.*

Theorem 2.5 generalizes various known results on substitutions. In particular, let us consider Theorem 2.5(1) for the language $K = \{x\}$ (i.e. the singleton language of the one letter word $x \in X^*$). The substitution $\sigma(x) = L$ is the only substitution satisfying $\sigma(K) = L$. Consequently, a substitution as Theorem 2.5(1) exists iff $\mathsf{sh}(L) \leq h$, and henceforth, Theorem 2.5(1) includes a solution to the star height problem.

The *finite substitution problem* means to decide for given recognizable languages K and L whether there is a finite substitution σ such that $\sigma(K) = L$. It was already raised by J.-E. Pin around 1992 [49]. S. Bala showed that this problem is EXPSPACE-complete [1, 2]. The author showed independently the decidability of a slightly weaker variant of this problem [22, 25]. Remarkably, both S. Bala and the author independently introduced the notion of a desert automaton to decide this problem [1, 2, 22, 25].

A substitution σ is finite iff $\mathsf{sh}(\sigma) = 0$. Hence, Theorem 2.5 includes a solution to the finite substitution problem. S. Bala's algorithm for the finite substitution problem requires just single exponential space [1] while Theorem 2.5 gives just a double exponential space bound. However, for $h = 0$, we can optimize our approach to single exponential space (cf. Section 7.4).

Similarly to Theorem 2.5, one can ask for a recognizable (resp. arbitrary) substitution σ such that $\sigma(K) = L$ (*recognizable substitution problem* resp. *substitution problem*). S. Bala showed that these problems are EXPSPACE-complete [1]. In fact, one can show that there exists a (not necessarily recognizable) substitution σ satisfying $\sigma(K) = L$ iff there exists a recognizable substitution σ' such that $\sigma'(K) = L$ and $\mathsf{sh}(\sigma') \leq 2^{|Q_L|^2}$ whereas Q_L are the states of a nondeterministic automaton for L (cf. Lemma 7.1). Consequently, Theorem 2.5 includes solutions to the recognizable substitution problem and the substitution problem. Again, S. Bala's algorithms require just single exponential space [1] while Theorem 2.5 gives just a double exponential space bound. Again, we can optimize our approach to single exponential space (cf. Section 7.4).

2.6 Bibliographic Remarks

The notions in Section 2.1 and 2.2 are classic. The notions in Section 2.3 originate from the author. The notion of a $\sharp$-expression was introduced K. Hashiguchi [15]. Distance automata were introduced by K. Hashiguchi in 1982 [11]. Desert automata were developed independently by S. Bala and the author in 2003 [1, 22, 25]. The notion of a nested distance desert automaton originates from the author as a joint generalization and further development of distance automata and desert automata.

Theorem 2.1 originates from the author, but it was already known for distance automata. In the particular case of distance automata, (1) $\Leftrightarrow$ (2) was shown by H. Leung and I. Simon [29, 30, 53], (1) $\Leftrightarrow$ (3) was shown by K. Hashiguchi [15], and (1) $\Leftrightarrow$ (2) $\Leftrightarrow$ (3) was shown by H. Leung and I. Simon [53, 34]. However, the bound on the $\sharp$-height of r in (3) was unknown even for distance automata!

Theorem 2.2 originates from the author, but it was already known for distance automata [35] and desert automata [1, 22]. The proof of the decidability in PSPACE is not a generalization of these two particular cases but a new approach which answers a question from H. Leung's PhD thesis from 1987 [29].

Theorem 2.3 originates from the author, but the decidability was already shown by K. Hashiguchi [14] without a bound on the complexity. Theorem 2.4 seems to be well-known in the community but the author did not find any reference. Its proof was shown by the author.

Theorem 2.5 originates from the author. The particular case $h = 0$ in Theorem 2.5 was already shown by S. Bala and in a slightly weaker way the author [1, 23].

3 Some Algebraic and Technical Foundations

We develop some algebraic and technical foundations which are required in Sections 3, 4, and 5.

In Section 3.1, we get familiar with classic ideas from ideal theory. In Section 3.2, we introduce the notion of a consistent mapping as an abstraction of several particular mappings used by I. Simon and H. Leung. In Section 3.3, we develop a semiring to describe nested distance desert automata in an algebraic fashion. In Section 3.4, we show some technical lemmas about connections between the concatenation and iteration of words in V^+ and their weights.

In order to understand Sections 3, 4, and 5, it suffices to read Section 3.1 very briefly, and Sections 3.2 and 3.4 briefly. However, a good understanding of the ideas and constructions in Section 3.3 is necessary.

3.1 Ideal Theory of Finite Semigroups

We introduce some concepts from ideal theory. This section is far away from being a comprehensive overview, for a deeper understanding, the author recommends textbooks, e.g., [10, 28, 44].

To be self-contained, we present this section in complete detail. However, to understand the rest of this thesis it suffices to read Section 3.1 just briefly now, and to come back if necessary.

As already mentioned, a semigroup S is a set with a binary associative operation which we denote by juxtaposition. Let S be a semigroup within this section.

If there is no identity in S, then we denote by S^1 the semigroup consisting of the set $S \cup 1$, on which the operation of S is extended in a way that 1 is the identity of S^1. If S has an identity, then we define S^1 to be S.

We call an $e \in S$ an *idempotent* if $e^2 = e$. We denote the set of all idempotents of S by $\mathsf{E}(S)$.

The following relations are called Green's relation. We show several equivalent definitions. Let $a, b \in S$.

1. $a \leq_{\mathscr{J}} b \quad :\Longleftrightarrow \quad a \in S^1 b S^1 \quad \Longleftrightarrow \quad S^1 a S^1 \subseteq S^1 b S^1$
2. $a \leq_{\mathscr{L}} b \quad :\Longleftrightarrow \quad a \in S^1 b \quad \Longleftrightarrow \quad S^1 a \subseteq S^1 b$
3. $a \leq_{\mathscr{R}} b \quad :\Longleftrightarrow \quad a \in b S^1 \quad \Longleftrightarrow \quad a S^1 \subseteq b S^1$

We allow to denote $a \leq_{\mathscr{J}} b$ by $b \geq_{\mathscr{J}} a$, and similarly for the other relations. The relation $\leq_{\mathscr{L}}$ is right stable, and similarly, $\leq_{\mathscr{R}}$ is left stable. However, we do not have a similar property for $\leq_{\mathscr{J}}$.

Again, let $a, b \in S$. We define:

1. $a =_{\mathscr{J}} b \quad :\Longleftrightarrow \quad a \leq_{\mathscr{J}} b$ and $a \geq_{\mathscr{J}} b \quad \Longleftrightarrow \quad S^1 a S^1 = S^1 b S^1$
2. $a =_{\mathscr{L}} b \quad :\Longleftrightarrow \quad a \leq_{\mathscr{L}} b$ and $a \geq_{\mathscr{L}} b \quad \Longleftrightarrow \quad S^1 a = S^1 b$
3. $a =_{\mathscr{R}} b \quad :\Longleftrightarrow \quad a \leq_{\mathscr{R}} b$ and $a \geq_{\mathscr{R}} b \quad \Longleftrightarrow \quad a S^1 = b S^1$

It is easy to see that $=_{\mathscr{J}}$, $=_{\mathscr{L}}$, and $=_{\mathscr{R}}$ are equivalence relations. We call their equivalence classes $\mathscr{J}$-classes (resp. $\mathscr{L}$-, $\mathscr{R}$-classes). For every $a \in S$, we denote by $\mathscr{J}(a)$, $\mathscr{L}(a)$, resp. $\mathscr{R}(a)$ the $\mathscr{J}$-, $\mathscr{L}$-, resp. $\mathscr{R}$-class of a. As above, $=_{\mathscr{L}}$ resp. $=_{\mathscr{R}}$ are right stable resp. left stable.

Remark 3.1.
1. Let $e \in \mathsf{E}(S)$ and $a \leq_{\mathscr{L}} e$. There is some $p \in S^1$ such that $a = pe$. Hence, $ae = pee = pe = a$. Similarly, if $b \leq_{\mathscr{R}} e$, then $eb = b$.

2. Let $e, f \in \mathsf{E}(S)$ with $e \leq_{\mathscr{L}} f$ and $e \geq_{\mathscr{R}} f$. Then, $ef = e$ and $ef = f$, i.e., $e = f$.

On the set of idempotents $\mathsf{E}(S)$, one defines a natural ordering $\leq$ such that for every $e, f \in \mathsf{E}(S)$, we have $e \leq f$ iff $e = ef = fe$. By Remark 3.1, we have $e \leq f$ iff $e \leq_{\mathscr{L}} f$ and $e \leq_{\mathscr{R}} f$.

For every $a \in S^1$, we denote by $a\cdot$ resp. $\cdot a$ the left resp. right multiplication by a.

The following lemma due to J.A. Green is of crucial importance to understand the relations between $\mathscr{J}$-, $\mathscr{L}$-, and $\mathscr{R}$-classes.

Lemma 3.1. (Green's lemma). *Let S be a semigroup, $a, b \in S$ and $p, q \in S^1$.*

1. *If $b = ap$ and $a = bq$, then $\cdot p$ and $\cdot q$ are mutually inverse, $\mathscr{R}$-class preserving bijections between $\mathscr{L}(a)$ and $\mathscr{L}(b)$.*

2. *If $b = pa$ and $a = qb$, then $p\cdot$ and $q\cdot$ are mutually inverse, $\mathscr{L}$-class preserving bijections between $\mathscr{R}(a)$ and $\mathscr{R}(b)$.*

The notion *$\mathscr{R}$-class preserving* (and similarly *$\mathscr{L}$-class preserving*) means that we have $c =_{\mathscr{R}} cp$ for every $c \in \mathscr{L}(a)$ and $d =_{\mathscr{R}} dq$ for every $d \in \mathscr{L}(b)$.

Proof. We show (1). Let $c \in \mathscr{L}(a)$. We have $b = ap =_{\mathscr{L}} cp$, because $=_{\mathscr{L}}$ is right stable. Thus, $cp \in \mathscr{L}(b)$. There is an $x \in S^1$ such that $c = xa$. By $apq = a$, we have $xapq = xa$, i.e., $cpq = c$.

By $cp \in \mathscr{L}(b)$ and $cpq = c$ for every $c \in \mathscr{L}(a)$, we know that $\cdot pq$ is the identity on $\mathscr{L}(a)$, and moreover, $\cdot p : \mathscr{L}(a) \to \mathscr{L}(b)$ is injective and $\cdot q : \mathscr{L}(b) \to \mathscr{L}(a)$ is surjective. In a symmetric way, we can show that $\cdot qp$ is the identity on $\mathscr{L}(b)$ and that $\cdot q : \mathscr{L}(b) \to \mathscr{L}(a)$ and $\cdot p : \mathscr{L}(a) \to \mathscr{L}(b)$ are injective resp. surjective. This completes (1). We can show (2) in a symmetric way. □

There are several connections between Green's relations and multiplication.

We assume from now that S is finite. Let $a \in S$. There $l, m \geq 1$ such that $a^l = a^{l+m}$. Then, $a^{2lm} = a^{lm} \in \mathsf{E}(S)$. Thus, for every $a \in S$ there is some $k \geq 1$ such that $a^k \in \mathsf{E}(S)$.

Lemma 3.2. *Let S be a finite semigroup. Let $a, b \in S$ and $p, q \in S^1$ be arbitrary.*

1. *If $a =_{\mathscr{J}} paq$, then $pa =_{\mathscr{L}} a =_{\mathscr{R}} aq$.*

2. *If $a =_{\mathscr{J}} ab$, then $a =_{\mathscr{R}} ab$.*

3. *If $b =_{\mathscr{J}} ab$, then $b =_{\mathscr{L}} ab$.*

4. *If $a =_{\mathscr{J}} b$, we have $\mathscr{L}(a) \cap \mathscr{R}(b) \neq \emptyset$.*

Proof.

1. By $a =_{\mathscr{J}} paq$, there are $r, s \in S^1$ such that $a = rpaqs$. Then, we have for every $k \geq 1$ $a = (rp)^k a (qs)^k$. Let $k \geq 1$ such that $(rp)^k \in \mathsf{E}(S)$. Then, $a = (rp)^k a (qs)^k = (rp)^k (rp)^k a (qs)^k = (rp)^k a$, and $a = (rp)^k a \leq_{\mathscr{L}} pa \leq_{\mathscr{L}} a$., i.e., $pa =_{\mathscr{L}} a$, and by symmetry, $a =_{\mathscr{R}} aq$.

2. By $a =_{\mathscr{J}} 1ab$ and (1), we have $a =_{\mathscr{R}} ab$.

3. By $b =_{\mathscr{J}} ab1$ and (1), we have $ab =_{\mathscr{L}} b$.

4. There are $p, q \in S^1$ such that $b = paq =_{\mathscr{J}} a$. By (1), we have $a =_{\mathscr{L}} pa$. By $paq =_{\mathscr{J}} a$, we have $paq =_{\mathscr{J}} pa$, and by (2), $paq =_{\mathscr{R}} pa$, i.e., $b =_{\mathscr{R}} pa$. Thus, $pa \in \mathscr{L}(a) \cap \mathscr{R}(b)$.

□

Lemma 3.2 cannot be generalized to infinite semigroups, although it is a rather challenging task to give a counter example without consulting a textbook [44].

There is another GREEN's relation. For every $a, b \in S$ let

1. $a =_{\mathscr{H}} b \quad :\Longleftrightarrow \quad a =_{\mathscr{L}} b$ and $a =_{\mathscr{R}} b$

Clearly, $=_{\mathscr{H}}$ is the intersection of $=_{\mathscr{L}}$ and $=_{\mathscr{R}}$. Hence, $=_{\mathscr{H}}$ is an equivalence relation and its equivalence classes ($\mathscr{H}$-classes) are the non-empty intersections of $\mathscr{L}$- and $\mathscr{R}$-classes.

Lemma 3.3. *Let S be a finite semigroup, H be a $\mathscr{H}$-class, and J be the $\mathscr{J}$-class with $H \subseteq J$. We have $HH \cap J \neq \emptyset$ iff H is a group.*

Proof. If H is a group, then we have $HH = H \subseteq J$, i.e., $HH \cap J = H \neq \emptyset$.

Conversely, assume $HH \cap J \neq \emptyset$. Let $p, q \in H$ satisfying $pq \in J$. Let $a, b \in H$ be arbitrary. By $a =_{\mathscr{L}} p$, we have $ab =_{\mathscr{L}} pb$. By $b =_{\mathscr{R}} q$, we have $pb =_{\mathscr{R}} pq$. Thus, $ab =_{\mathscr{J}} pq$, and by $a, b \in H$, we have $a =_{\mathscr{J}} b =_{\mathscr{J}} ab$. By Lemma 3.2(2,3), we have $ab \in \mathscr{R}(a) \cap \mathscr{L}(b)$, and by $a, b, p, q \in H$, we have $\mathscr{R}(a) = \mathscr{R}(p)$ and $\mathscr{L}(b) = \mathscr{L}(q)$. Consequently, $ab \in \mathscr{R}(p) \cap \mathscr{L}(q) = H$, i.e., H is closed under multiplication.

By $a =_{\mathscr{R}} ab$, there is some $x \in S^1$ such that $abx = a$. By Lemma 3.1 $\cdot b : H \to H$ and $\cdot x : H \to H$ are mutually inverse bijections. Similarly, there is some $y \in S^1$ such that $yab = b$, and $a\cdot : H \to H$ and $y\cdot : H \to H$ are mutually inverse bijections.

As seen above, there is some $k \geq 1$ such that $a^k \in \mathsf{E}(S)$. We denote $e := a^k$. Clearly, $e \in H$. Let $c \in H$ be arbitrary. Since, $c =_{\mathscr{H}} e$, we have $ec = ce = c$ by Remark 3.1(1), i.e., e is the identity in H. Since $\cdot c$ is a bijection on H, there is some $c' \in H$ such that $c'c = e$. □

The following lemma will be very useful.

Lemma 3.4. *Let S be a finite semigroup, let $a, b \in S$ satisfying $a =_{\mathscr{J}} b$. We have $a =_{\mathscr{J}} b =_{\mathscr{J}} ab$ iff there is an idempotent $e \in \mathsf{E}(S)$ such that $a =_{\mathscr{L}} e =_{\mathscr{R}} b$.*

Proof. Let $e \in \mathsf{E}(S)$ such that $a =_{\mathscr{L}} e =_{\mathscr{R}} b$. There are $x, y \in S^1$ satisfying $xa = e = by$, i.e., $ab \geq_{\mathscr{J}} xaby = ee = e =_{\mathscr{J}} a$. Clearly, $ab \leq_{\mathscr{J}} a$. To sum up, $ab =_{\mathscr{J}} a$.

Conversely, assume $a =_{\mathscr{J}} b =_{\mathscr{J}} ab$. Let $H := \mathscr{L}(a) \cap \mathscr{R}(b)$ and $J := \mathscr{J}(a) = \mathscr{J}(b) = \mathscr{J}(ab)$. By Lemma 3.2(4), choose some $p \in H$. There are $x, y \in S^1$ satisfying $a = xp$ and $b = py$. Hence, $pp \geq_{\mathscr{J}} xppy = ab$. Moreover, $pp \leq_{\mathscr{J}} p =_{\mathscr{J}} a$. Consequently, $pp \in J$. By Lemma 3.3, H is a group. Let e be the identity of H. □

Usually, one visualizes a $\mathscr{J}$-class by an "egg-box picture" in which the columns are $\mathscr{L}$-classes and the **r**ows are $\mathscr{R}$-classes. We can combine Lemma 3.2(2,3,4) and Lemma 3.4: If $a =_{\mathscr{J}} b =_{\mathscr{J}} ab$, then $ab \in \mathscr{R}(a) \cap \mathscr{L}(b)$ and there is an idempotent $e \in \mathscr{L}(a) \cap \mathscr{R}(b)$ as shown in the following table:

a		ab	
e		b	

One distinguishes two kinds of $\mathscr{J}$-classes. If some $\mathscr{J}$-class J satisfies the three equivalent conditions in Lemma 3.5, then we call J a *regular* $\mathscr{J}$-class, otherwise we call J *non-regular*. We call some element $a \in S$ *regular*, if $\mathscr{J}(a)$ is a regular $\mathscr{J}$-class. We denote the set of all regular elements of S by $\mathsf{Reg}(S)$.

Lemma 3.5. *Let J be a $\mathscr{J}$-class of a finite semigroup S. The following assertions are equivalent:*

1. $JJ \cap J \neq \emptyset$
2. *There is at least one idempotent in J.*
3. *In every $\mathscr{L}$-class of J and in every $\mathscr{R}$-class of J there is at least one idempotent.*

Proof. (3) $\Rightarrow$ (2) and (2) $\Rightarrow$ (1) are obvious, and (1) $\Rightarrow$ (2) is an immediate consequence of Lemma 3.4.

We show (2) $\Rightarrow$ (3). Let $e \in J$ be an idempotent, and let $a \in J$ be arbitrary. We show that there is an idempotent in $\mathscr{L}(a)$. There are $p, q \in S^1$ such that $e = paq = (paq)^3 = pa(qpa)^2q$. We have

$$a \geq_{\mathscr{J}} qpa \geq_{\mathscr{J}} (qpa)^2 \geq_{\mathscr{J}} pa(qpa)^2q = (paq)^3 = e =_{\mathscr{J}} a,$$

i.e., $qpa =_{\mathscr{J}} (qpa)^2 \in J$. By Lemma 3.4, there is an idempotent in $\mathscr{L}(qpa) \cap \mathscr{R}(qpa)$. By Lemma 3.2(3). We have $\mathscr{L}(qpa) = \mathscr{L}(a)$, i.e., there is an idempotent in $\mathscr{L}(a)$.

By examining aqp, we can show in a symmetric way that there is an idempotent in $\mathscr{R}(a)$, and (3) follows from the arbitrary choice of a. ☐

Let T be a subsemigroup of S. We have $\mathsf{E}(T) = \mathsf{E}(S) \cap T$ and $\mathsf{Reg}(T) \subseteq \mathsf{Reg}(S)$. However, we do not necessarily have $\mathsf{Reg}(T) = \mathsf{Reg}(S) \cap T$.

The reader should be aware that in contrast to Lemma 3.3, a regular $\mathscr{J}$-class in not necessarily closed under multiplication. Even if a regular $\mathscr{J}$-class is closed under multiplication, then it is not necessarily a group.

The following property will be very useful.

Lemma 3.6. *Let S be a finite semigroup and let $a, b \in S$ such that $a =_{\mathscr{J}} b$.*
If $ab = a$, then $b \in \mathsf{E}(S)$. If $ab = b$, then $a \in \mathsf{E}(S)$.

Proof. Assume $ab = a$. Then, $ab = a =_{\mathscr{J}} b$, and by Lemma 3.2(3), we have $ab =_{\mathscr{L}} b$, i.e., $a =_{\mathscr{L}} b$. Hence, there is some $p \in S^1$ such that $pa = b$. Thus, $pab = pa$, i.e., $b^2 = b \in \mathsf{E}(S)$. The other assertion follows by symmetry. ☐

The assumption $a =_{\mathscr{J}} b$ in Lemma 3.6 is crucial. Just assume that S has a zero and consider the case $a = 0$.

The next lemma is well-known in semigroup theory and of importance in the theory of recognizable languages [44].

Lemma 3.7. *Let S be a finite semigroup. Let $e, f \in \mathsf{E}(S)$ satisfying $e =_{\mathscr{J}} f$.*
For every $a \in \mathscr{R}(e) \cap \mathscr{L}(f)$, there is exactly one $b \in \mathscr{R}(f) \cap \mathscr{L}(e)$ satisfying both $ab = e$ and $ba = f$.

We can visualize the relations between a, b, e, f in Lemma 3.6 by the following egg-box picture:

a		e	
f		b	

Proof. Let a, e, f as in the lemma. There are $p, q \in S^1$ such that $ap = e$ and $apq = eq = a$. Moreover, there are $x, y \in S^1$ such that $xa = f$ and $yxa = yf = a$.

By Lemma 3.1, $\cdot p$ and $\cdot q$ are mutually inverse bijections between $\mathscr{L}(a) = \mathscr{L}(f)$ and $\mathscr{L}(e)$. Similarly, $x\cdot$ and $y\cdot$ are mutually inverse bijections between $\mathscr{R}(a) = \mathscr{R}(e)$ and $\mathscr{R}(f)$.

The crucial fact is that we have $xap = xe$ but also $xap = fp$. We set $b := xe = fp$. By Lemma 3.1, we have $b \in \mathscr{R}(f) \cap \mathscr{L}(e)$.

By $a = xf$ and $f \in \mathsf{E}(S)$, we have $af = a$, and by symmetry $ea = a$.

We have $ab = afp = ap = e$ and $ba = xea = xa = f$.

Let $b' \in \mathscr{R}(f) \cap \mathscr{L}(e)$ such that $ab' = e$ and $b'a = f$. By $ab' = e$, we have $xab' = xe$, and thus, $fb' = b$. However, by $f =_{\mathscr{R}} b'$, we have $fb' = b'$, i.e. $b' = b$. □

For every $k > 0$ and $a_1, \ldots, a_k \in S$, we call $a_1, \ldots, a_k$ a *smooth product* if we have $a_1 =_{\mathscr{J}} a_2 =_{\mathscr{J}} \ldots =_{\mathscr{J}} a_k =_{\mathscr{J}} (a_1 \ldots a_k) \in \mathsf{Reg}(S)$. Note that this is not a classic notion.

Let J_1 and J_2 be two $\mathscr{J}$-classes. There are $a \in J_1$ and $b \in J_2$ satisfying $a \leq_{\mathscr{J}} b$ iff we have $a \leq_{\mathscr{J}} b$ for every $a \in J_1$ and $b \in J_2$. Hence, $\leq_{\mathscr{J}}$ extends to a partial ordering of the $\mathscr{J}$-classes.

In a finite semigroup, there is always a maximal $\mathscr{J}$-class, but it is not necessarily unique.

We call some subset of $I \subseteq S$ an *ideal* if $S^1 I S^1 \subseteq I$. Obviously, some subset $I \subseteq S$ is an ideal iff I is closed under $\leq_{\mathscr{J}}$, i.e., iff for every $a \in S$, $b \in I$ with $a \leq_{\mathscr{J}} b$ we have $a \in I$. In particular, every ideal of S is saturated by the $\mathscr{J}$-classes of S.

If S is finite, then there are some $z \geq 1$ and ideals $I_1, \ldots, I_{z+1}$ of S satisfying

$$S = I_1 \supsetneq I_2 \supsetneq \cdots \supsetneq I_z \supsetneq I_{z+1} = \emptyset$$

such that for every $l \in \{1, \ldots, z\}$, the set $I_l \setminus I_{l+1}$ is a $\mathscr{J}$-class. Moreover, z is the number of $\mathscr{J}$-classes of S. It is easy to construct such a chain of ideals: you simply start by $I_1 := S$, then we set $I_2 := I_1 \setminus J_1$ where J_2 is a maximal $\mathscr{J}$-class and so on.

This closes our expedition to the realms of ideal theory. The reader should be aware that ideal theory is just an initial part of the huge field of the structure theory of semigroups. Moreover, the notions and results in this section are just the beginning of ideal theory, and there are many important aspects which are not covered here. For example, there is a deep theorem by D. Rees and A.K. Sushkevich which describes the inner structure of regular $\mathscr{J}$-classes of finite semigroups up to isomorphism.

3.2 Consistent Mappings

We develop the notion of a consistent mapping as an abstraction from certain transformations (stabilization, perforation) of matrices over various semirings which play a key role in many articles by I. Simon and H. Leung [29, 30, 31, 32, 34, 51, 52, 53]. The notions and results in this section are of crucial importance to understand Section 4 and 5, but the proofs can be skipped.

Let S be a finite semigroup.

We call a mapping $\sharp : \mathsf{E}(S) \to \mathsf{E}(S)$ *consistent*, if for every $a, b \in S^1$ and $e, f \in \mathsf{E}(S)$ with $e =_{\mathscr{J}} f$ and $f = aeb$, we have $f^\sharp = ae^\sharp b$.

If $\sharp$ is a consistent mapping and $e \in \mathsf{E}(S)$, then $e = 1ee = ee1 = eee$, and thus, $e^\sharp = e^\sharp e = ee^\sharp = ee^\sharp e$, i.e., $e^\sharp \leq_{\mathscr{L}} e$, and $e^\sharp \leq_{\mathscr{R}} e$. Thus, $e^\sharp \leq e$ in the natural ordering $\leq$ of the idempotents.

We use some results from finite semigroup theory to show that every consistent mapping admits a unique extension to regular elements. Lemma 3.8 was already shown by H. Leung in a more particular framework [29, 30, 34].

Lemma 3.8. *Let $\sharp$ be a consistent mapping. Let $e, f \in \mathsf{E}(S)$ and $a, b, c, d \in S^1$ satisfying $aeb = cfd =_{\mathscr{J}} e =_{\mathscr{J}} f$. We have $ae^\sharp b = cf^\sharp d$.*

Proof. Let J be the $\mathscr{J}$-class with $aeb = cfd =_{\mathscr{J}} e =_{\mathscr{J}} f \in J$. We have $ae, eb, rf, fd \in J$. As seen in Section 3.1, $ae =_{\mathscr{R}} aeb = cfd =_{\mathscr{R}} cf$ and $eb =_{\mathscr{L}} aeb = cfd =_{\mathscr{L}} fd$.

ae		cf		$aeb = cfd$
		f		fd
e				eb

There is some $p \in S^1$ such that $pcf = f$. By GREEN's lemma (Lemma 3.1), $p\cdot$ and $c\cdot$ are mutually inverse bijections between $\mathscr{R}(cf)$ and $\mathscr{R}(f)$. By $ae \in \mathscr{R}(cf)$, we have $ae = cpae$.

Similarly, there is some $r \in S^1$ such that $fdr = f$ and $eb = ebrd$.

By $cfd = aeb$, we have $pcfdr = paebr$, and thus, $f = paebr$. We have $f^\sharp = pae^\sharp br$. Then, we have $ff^\sharp f = paee^\sharp ebr$, and $cff^\sharp fd = cpaee^\sharp ebrd$, i.e., $cff^\sharp fd = aee^\sharp eb$, and finally, $cf^\sharp d = ae^\sharp b$. □

Lemma 3.8 allows us to extend consistent mappings to regular elements of S.

Corollary 3.9. *Let $\sharp : \mathsf{E}(S) \to \mathsf{E}(S)$ be a consistent mapping. By setting $(aeb)^\sharp := ae^\sharp b$ for every $a, b \in S^1$, $e \in \mathsf{E}(S)$ satisfying $e =_{\mathscr{J}} aeb$, we define a mapping $\sharp : \mathsf{Reg}(S) \to \mathsf{Reg}(S)$.*

Proof. The mapping $\sharp : \mathsf{Reg}(S) \to \mathsf{Reg}(S)$ is well-defined by Lemma 3.8. It remains to show $ae^\sharp b \in \mathsf{Reg}(S)$. By $aeb =_{\mathscr{J}} e$, we have $ae =_{\mathscr{L}} e =_{\mathscr{R}} eb$. There are $c, d \in S^1$ such that $cae = e = ebd$. Thus, $cae^\sharp bd = caee^\sharp ebd = ee^\sharp e = e^\sharp$, and thus, $ae^\sharp b =_{\mathscr{J}} e^\sharp$, i.e., $e^\sharp$ is an idempotent in $\mathscr{J}(ae^\sharp b)$. □

Remark 3.2. *Let $a \in S$ be arbitrary and $e, f \in S$ satisfying $e =_{\mathscr{R}} a =_{\mathscr{L}} f$. Then, $ea = af = a$ and $e^\sharp a = af^\sharp = a^\sharp$. Consequently, $a^\sharp \leq_{\mathscr{L}} a$ and $a^\sharp \leq_{\mathscr{R}} a$.*

The next lemma allows to deal with consistent mappings in a very convenient way.

Lemma 3.10. *Let $a, b, c \in S^1$.*

1. *If $abc =_{\mathscr{J}} b \in \mathsf{Reg}(S)$, then we have $(abc)^\sharp = ab^\sharp c$.*
2. *If $a =_{\mathscr{J}} b =_{\mathscr{J}} ab \in \mathsf{Reg}(S)$, then we have $(ab)^\sharp = a^\sharp b = ab^\sharp = a^\sharp b^\sharp$.*

Proof. (1) Because $b \in \mathsf{Reg}(S)$, there is some $e \in \mathsf{E}(S)$ with $e =_{\mathscr{L}} b$, i.e., $be = b$. By the extension of $\sharp$, we have $b^\sharp = be^\sharp$ and $ab^\sharp c = abe^\sharp c$. For $(abc)^\sharp$, we obtain $(abc)^\sharp = (abec)^\sharp = abe^\sharp c$.

(2) There is some $e \in \mathsf{E}(S)$ such that $a =_{\mathscr{L}} e =_{\mathscr{R}} b$, i.e., $ae = a$ and $eb = b$.
We have $(ab)^\sharp = ae^\sharp b = (ae)^\sharp b = a^\sharp b$, $(ab)^\sharp = ae^\sharp b = a(eb)^\sharp = ab^\sharp$, and $(ab)^\sharp = ae^\sharp b = ae^\sharp e^\sharp b = (ae)^\sharp (eb)^\sharp = a^\sharp b^\sharp$. □

Consider in the case $c = 1$ in (1). Then, $(ab)^\sharp = ab^\sharp$. Similarly, $(bc)^\sharp = b^\sharp c$, if $a = 1$.

If $a, b, c \in S$ are a smooth product, then we can play with a consistent mapping:

$$(abc)^\sharp = a^\sharp bc = ab^\sharp c = abc^\sharp = a^\sharp b^\sharp c = a^\sharp bc^\sharp = ab^\sharp c^\sharp = a^\sharp b^\sharp c^\sharp = (ab)^\sharp c^\sharp = \ldots$$

For the consistent mappings (stabilizations) used by I. SIMON and H. LEUNG we have $e^\sharp = (e^\sharp)^\sharp$ for every $e \in \mathsf{E}(S)$. However, this property does not hold for every consistent mapping, as the following example shows.

Example 3.1. Consider the monoid over $M = \{1, \ldots, 9\}$ with the maximum operation defined by the usual ordering of the integers. It is easy to verify that the mapping defined by $x^\sharp := x + 1$ for $x \in \{1, \ldots, 8\}$ and $9^\sharp = 9$ is consistent. However, we have, e.g., $2^\sharp = 3 \neq 4 = (2^\sharp)^\sharp$. $\square$

Let us mention that there is a characterization of consistent mappings [25]: A mapping $\sharp : \mathsf{E}(S) \to \mathsf{E}(S)$ is consistent iff for every $a, b \in S^1$ with $ab, ba \in \mathsf{E}(S)$, we have $(ab)^\sharp = a(ba)^\sharp b$.

3.3 The Nested Distance Desert Semiring

In this section, we develop a semiring $\mathcal{V}$ to describe nested distance desert automata in an algebraic way. In particular, we use matrices over $\mathcal{V}$ as transformation matrices. Recall that we defined $V := \{\angle_0, \curlyvee_0, \angle_1, \curlyvee_1, \ldots, \curlyvee_{h-1}, \angle_h\}$.

Let $h \in \mathbb{N}$. Let $\mathcal{V} = V \cup \{\omega, \infty\}$ and consider the ordering

$$\angle_0 \sqsubseteq \curlyvee_0 \sqsubseteq \angle_1 \sqsubseteq \curlyvee_1 \sqsubseteq \ldots \sqsubseteq \curlyvee_{h-1} \sqsubseteq \angle_h \sqsubseteq \omega \sqsubseteq \infty$$

on $\mathcal{V}$. We define a multiplication $\cdot$ on $\mathcal{V}$ as the maximum for $\sqsubseteq$. Let $\psi : V^+ \to \mathcal{V}$ be the canonical homomorphism.

Let $\pi \in V^+$. We say that we can *walk along* π *in a cycle* iff there is some $d \in \mathbb{N}$ such that for every $k \in \mathbb{N}$, we have $\Delta(\pi^k) \leq d$. We show that we can walk along π in a cycle iff $\psi(\pi) \in \{\curlyvee_0, \ldots, \curlyvee_{h-1}\}$. This is a key property of ψ. Indeed, assume that $\psi(\pi) = \angle_g$ for some $0 \leq g \leq h$. Then, π contains the letter $\angle_g$, i.e., we have to pay an g-coin when we walk along π. By the definition of ψ, π does not contain $\curlyvee_g, \angle_{g+1}, \ldots, \curlyvee_{h-1}, \angle_h$. Thus, we cannot obtain g-coins in π. Hence, for every $k \in \mathbb{N}$, $\Delta(\pi^k) \geq k$, i.e., we cannot walk along π in a cycle. Conversely, if $\psi(\pi) = \curlyvee_g$ for some $0 \leq g < h$, then can walk along π in a cycle, because we can obtain 0-coins, $\ldots$, g-coins and we do not have to pay $(g+1)$-coins, $\ldots$, h-coins along π. As a conclusion, the set of all words $\pi \in V^+$ along which we can walk in a cycle is a recognizable language of V^+ and $\psi : V^+ \to \mathcal{V} \setminus \{\omega, \infty\}$ is its syntactic homomorphism.

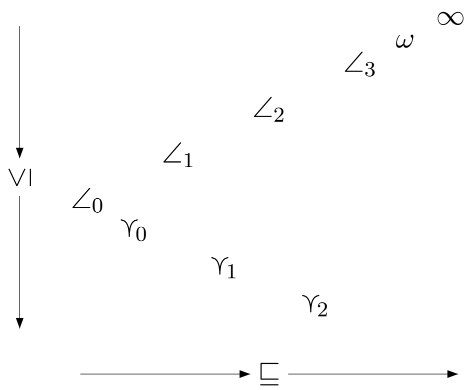
An extension of ψ to V^* is only possible by setting $\psi(\varepsilon) = \angle_0$, since otherwise, ψ is no longer a homomorphism. However, for every $k \in \mathbb{N}$, we have $\Delta(\varepsilon^k) = 0$. Hence, we do not have any longer the key property that we can walk along some path π in a cycle iff $\psi(\pi) \in \{\curlyvee_0, \ldots, \curlyvee_{h-1}\}$. Consequently, we rather leave $\psi(\varepsilon)$ undefined.

Now, consider the following ordering on $\leq$ on $\mathcal{V}$, which differs from $\sqsubseteq$:

$$\curlyvee_{h-1} \leq \curlyvee_{h-2} \leq \ldots \leq \curlyvee_0 \leq \angle_0 \leq \ldots \leq \angle_h \leq \omega \leq \infty. \tag{1}$$

Intuitively, $\leq$ reflects which transitions we prefer. Given the choice between two transitions marked resp. by $\curlyvee_g$ and $\curlyvee_{g-1}$ (for some $0 < g < h$), then we choose the transition marked by $\curlyvee_g$, because $0, \ldots, g$-coins can be obtained at $\curlyvee_g$, but just $0, \ldots, (g-1)$-coins can be obtained at $\curlyvee_{g-1}$. Given the choice between two transitions marked resp. by $\angle_g$ and $\angle_{g+1}$ (for some $0 \leq g < h$), then we choose the transition marked by $\angle_g$, because $(g+1)$-coins are considered as more valuable than g-coins. We define an operation min on $\mathcal{V}$ as the minimum for $\leq$.

The following figure shows the relations $\sqsubseteq$ and $\leq$ for $h = 3$, where $\sqsubseteq$ corresponds to "left of" and $\leq$ corresponds to "below".



Let $z, z' \in \mathcal{V}$. If $z \sqsubseteq z'$ and $z \neq z'$, then we write $z \sqsubset z'$. We write $z < z'$ if $z \leq z'$ and $z \neq z'$.

Remark 3.3. Let $0 \leq g \leq h$. For every $z \sqsubseteq \angle_g$, we have $z \in \{\curlyvee_0, \ldots, \curlyvee_{g-1}, \angle_0, \ldots, \angle_g\}$, and thus, $z \leq \angle_g$. Similarly, $z \sqsubset \angle_g$ implies $z < \angle_g$.

Next, we show that the ordering $\leq$ on $\mathcal{V}$ is stable w.r.t. multiplication. We multiply the entire chain (1) by every member of $\mathcal{V}$. If we multiply (1) by ω (resp. ∞), then we obtain $\omega \leq \cdots \leq \omega \leq \infty$ (resp. $\infty \leq \cdots \leq \infty$), which is true. It is easy to see that (1) remains true if we multiply every element by $\curlyvee_g$ for $0 \leq g < h$ or by $\angle_g$ for $0 \leq g \leq h$.

As a consequence, for every $x, y, x', y' \in \mathcal{V}$ with $x \leq x'$ and $y \leq y'$, we have $xy \leq x'y \leq x'y'$.

Consequently, multiplication $\cdot$ on $\mathcal{V}$ distributes over min. Obviously, min and $\cdot$ are associative and commutative. Moreover, ∞ is a zero for $\cdot$ and an identity for min. Finally, $\angle_0$ is an identity for $\cdot$. Consequently, $(\mathcal{V}, \mathsf{min}, \cdot)$ is a commutative semiring which we call the *h-nested distance desert semiring*. We denote by $\mathcal{V}_{n\times n}$ the semiring of $n\times n$-matrices over $\mathcal{V}$.

For every $a, b \in \mathcal{V}_{n\times n}$ and every i, l, j, we have $(ab)[i,j] = \mathsf{min}_{1\leq k\leq n} a[i,k]\cdot b[k,j] \leq a[i,l]\cdot b[l,j]$.

Let $a, b \in \mathcal{V}_{n\times n}$. We denote $a \approx b$ if for every i, j we have $a[i,j] = \infty$ iff $b[i,j] = \infty$. It is straightforward to verify that $\approx$ is a congruence relation on $\mathcal{V}_{n\times n}$ and $\mathcal{V}_{n\times n}/\!\approx$ is isomorphic to the semiring of $n\times n$-matrices over the boolean semiring.

We close this section by a useful lemma for idempotent matrices.

Lemma 3.11. *Let $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ and i, j be arbitrary.*
There is some l such that $e[i,j] = e[i,l]\cdot e[l,l]\cdot e[l,j]$.

Proof. For every l, we have

$$e[i,j] = e^3[i,j] = \mathsf{min}_{1\leq k,k'\leq n}\big(e[i,k]\cdot e[k,k']\cdot e[k',j]\big) \leq e[i,l]\cdot e[l,l]\cdot e[l,j].$$

Since $e = e^{n+2}$, there are $i = i_0, \ldots, i_{n+2} = j$ such that $e[i,j] = e[i_0,i_1]\cdots e[i_{n+1},i_{n+2}]$. By a counting argument, there are $1 \leq p < q \leq (n+1)$ such that $i_p = i_q$. Let $l := i_p$. We have $e[i,l] = e^p[i,l] \leq e[i_0,i_1]\cdots e[i_{p-1},i_p]$, $e[l,l] = e^{q-p}[l,l] \leq e[i_p,i_{p+1}]\cdots e[i_{q-1},i_q]$, and $e[l,j] = e^{n+2-q}[l,j] \leq e[i_q,i_{n+2}]\cdots e[i_{n+1},i_{n+2}]$. Hence,

$$e[i,l]\cdot e[l,l]\cdot e[l,j] \leq e[i_0,i_1]\cdots e[i_{n+1},i_{n+2}] = e[i,j]$$

and the claim follows. □

Let $\mathcal{A} = [Q, E, I, F, \theta]$ be an h-nested distance desert automaton. Let $n := |Q|$ and assume $Q = \{1, \ldots, n\}$. We define a mapping $\Psi : \Sigma^+ \to \mathcal{V}_{n\times n}$ by setting for every $w \in \Sigma^+$, i, j

$$\Psi(w)[i,j] := \mathsf{min}_{\pi \in i \overset{w}{\rightsquigarrow} j}\, \psi(\theta(\pi)).$$

It is well-known in the theory of weighted automata that Ψ is a homomorphism. It will be of crucial importance for the decidability limitedness.

Let us mention that the semiring of $\mathcal{V}$ over the set $\mathcal{R} = \{\Upsilon_0, \angle_1, \omega, \infty\}$ was used by I. Simon and H. Leung to show the decidability of limitedness of distance automata [29, 30, 51, 53, 32, 34]. Similarly, the semiring of $\mathcal{V}$ over the set $\mathcal{D} = \{\angle_0, \Upsilon_0, \omega, \infty\}$ was used by the author to show the decidability of limitedness of desert automata [22, 25].

3.4 On the Weights of Words

As in Section 2.3, let $h \geq 0$ be arbitrary and $V := \{\angle_0, \Upsilon_0, \angle_1, \Upsilon_1, \ldots, \Upsilon_{h-1}, \angle_h\}$. We show some technical lemmas about the effects of the concatenation of words over V and their weights.

Lemma 3.12. *For every $\pi_1, \pi_2 \in V^+$, we have* $\max\{\Delta(\pi_1), \Delta(\pi_2)\} \leq \Delta(\pi_1\pi_2) \leq \Delta(\pi_1) + \Delta(\pi_2)$.

Proof. We have $\Delta(\pi_1) \leq \Delta(\pi_1\pi_2)$ and $\Delta(\pi_2) \leq \Delta(\pi_1\pi_2)$, because every factor of π_1 resp. π_2 is also a factor of $\pi_1\pi_2$. We can easily show $\Delta(\pi_1\pi_2) \leq \Delta(\pi_1) + \Delta(\pi_2)$, because every factor of $\pi_1\pi_2$ is a concatenation of a factor of π_1 and a factor of π_2. □

The bounds in Lemma 3.12 are sharp, just consider $\pi_1 := \pi_2 := \angle_0\Upsilon_0$ (resp. $\pi_1 := \pi_2 := \angle_0\angle_0$).

Lemma 3.13.

1. *Let $\pi \in V^+$ with $\psi(\pi) \in \{\angle_0, \ldots, \angle_h\}$. For every $k \geq 1$, we have $\Delta(\pi^k) \geq k$.*
2. *Let $\pi \in V^+$ with $\psi(\pi) \in \{\Upsilon_0, \ldots, \Upsilon_{h-1}\}$. For every $k \geq 1$, we have $\Delta(\pi^k) \leq 2\Delta(\pi)$ and $\Delta(\pi^k) < |\pi|$.*

Proof. (1) Let $0 \leq g \leq h$ such that $\psi(\pi) = \angle_g$. We have $\pi^k \in \{\angle_0, \Upsilon_0, \ldots, \angle_g\}^+$, and $|\pi^k|_g \geq k$, and thus, $|\Delta(\pi^k)| \geq k$.

(2) Let $0 \leq g < h$ such that $\psi(\pi) = \Upsilon_g$. For every $g < g' \leq h$, we have $|\pi^k|_{g'} = 0$. Now, let $0 \leq g' \leq g$, and let π' be a factor of π^k with $\pi' \in \{\angle_0, \Upsilon_0, \ldots, \Upsilon_{g'-1}, \angle_{g'}\}^*$. Because Υ_g occurs in π but not in π', we can factorize π' as $\pi' = \pi_1\pi_2$ for factors π_1, π_2 of π. We have

$$|\pi'|_{g'} = |\pi_1|_{g'} + |\pi_2|_{g'} \leq 2\Delta(\pi).$$

Thus, $\Delta(\pi^k) \leq 2\Delta(\pi)$. We easily see $|\pi'| < |\pi|$, i.e., $|\pi'|_{g'} < |\pi|$, and thus, $\Delta(\pi^k) < |\pi|$. □

The bounds in Lemma 3.13 are sharp: for (1), let $\pi = \Upsilon_0\angle_1\Upsilon_0$, and for (2), let $\pi = \angle_0\angle_0\Upsilon_0\angle_0\angle_0$.

Let $\mathbb{R}_+$ be the positive real numbers. For every $g \in \mathbb{N}$, we define a mapping $f_g : \mathbb{R}_+ \to \mathbb{R}_+$ by $f_g(x) := \sqrt[g+1]{x+1} - 1$ for $x \in \mathbb{R}_+$.

Lemma 3.14. *For every $g \in \mathbb{N}$ and $x \in \mathbb{R}_+$, we have* $f_{g+1}(x) = f_g\left(\dfrac{x - f_{g+1}(x)}{f_{g+1}(x) + 1}\right)$.

Proof. We have $\big(f_{g+1}(x) + 1\big)^{g+2} = x + 1$, and thus, $\big(f_{g+1}(x) + 1\big)^{g+1} = \dfrac{x+1}{f_{g+1}(x) + 1}$, and further,

$$f_{g+1}(x) = \sqrt[g+1]{\frac{x+1}{f_{g+1}(x)+1}} - 1 = \sqrt[g+1]{\frac{x - f_{g+1}(x)}{f_{g+1}(x)+1} + 1} - 1 = f_g\left(\frac{x - f_{g+1}(x)}{f_{g+1}(x)+1}\right).$$

□

Lemma 3.15. *Let $0 \le g \le h$ be arbitrary.*
For every $k \ge 1$ and $\pi_1, \ldots, \pi_k \in V^+$ with $\psi(\pi_1 \ldots \pi_k) = \angle_g$ and $\psi(\pi_l) \in \{\angle_0, \ldots, \angle_h\}$ for every $0 \le l \le k$, we have $\Delta(\pi_1 \ldots \pi_k) \ \ge\ f_g(k) \ =\ \sqrt[g+1]{k+1} - 1$.

At first, we sketch the idea to prove Lemma 3.15. For example, let $g = 2$ and $k = 124$, and $\pi_1 \ldots, \pi_{124} \in V^+$ as in the lemma. We denote $\pi := \pi_1 \ldots \pi_{124}$. We have to show $\Delta(\pi) \ge 4$.

Let $P := \{\, l \mid 1 \le l \le 124,\ \psi(\pi_l) = \angle_2 \,\}$. We have $\Delta(\pi) \ge |P|$. If $|P| \ge 4$, then we are done.

Now, assume, e.g., $P = \{10, 47, 93\}$. We consider the maximal interval[2] of $\{1, \ldots, 124\} \setminus P$, i.e., we consider the set $\{48, \ldots, 92\}$ and examine the word $\pi' := \pi_{48} \ldots \pi_{92}$. We have $\psi(\pi') \in \{\angle_0, \angle_1\}$ by the definition of P and the assumptions on $\pi_1 \ldots, \pi_{124}$. By induction on g, we assume that the lemma is true for π'. Thus, we have $\Delta(\pi') \ \ge\ \sqrt[1]{46} - 1 \ =\ 45$ or $\Delta(\pi') \ \ge\ \sqrt[2]{46} - 1 \ >\ 5$.

There are indeed $\pi_1 \ldots, \pi_{124} \in V^+$ with the above properties and $\Delta(\pi_1 \ldots \pi_{124}) = 4$. Just let $\pi'_1 := {\angle_0}^4$, $\pi'_2 := (\pi'_1 \angle_1)^4 \pi'_1$, and $\pi := (\pi'_2 \angle_2)^4 \pi'_2$, and let $\pi_1, \ldots, \pi_{124}$ be the letters of π. However, for $g = 2$ and $k = 125$, Lemma 3.15 shows $\Delta(\pi_1 \ldots \pi_{125}) \ge 5$.

Proof of Lemma 3.15. We show the lemma by an induction on g. At first, assume $g = 0$. We have $\pi_1 \ldots \pi_k \in {\angle_0}^+$ and $|\pi_1 \ldots \pi_k| \ge k$. Thus, $\Delta(\pi) \ge k = \sqrt[1]{k+1} - 1$.

Let $0 \le g < h$. By induction, we assume that the claim is true for $0, \ldots, g$, and we show the claim for $g + 1$. Choose some $k \ge 1$ and $\pi_1, \ldots, \pi_k \in V^+$ as in the lemma. Denote $\pi := \pi_1 \ldots \pi_k$. Let $P := \{\, l \mid 1 \le l \le k,\ \psi(\pi_l) = \angle_{g+1} \,\}$. If $|P| \ \ge\ f_{g+1}(k)$, then we have $\Delta(\pi) \ge f_{g+1}(k)$, because $\pi \in \{\angle_0, \curlyvee_0, \ldots, \angle_{g+1}\}^*$. We assume $|P| \ <\ f_{g+1}(k)$ in the rest of the proof.

We estimate the average cardinality of the maximal intervals of $\{1, \ldots, k\} \setminus P$. There are at least $k - |P| \ >\ k - f_{g+1}(k)$ members in $\{1, \ldots, k\} \setminus P$. On the other hand, there are at most $|P| + 1$, i.e., at most $f_{g+1}(k)$ maximal intervals in $\{1, \ldots, k\} \setminus P$. Thus, the average cardinality of the maximal intervals of $\{1, \ldots, k\} \setminus P$ is at least

$$\frac{k - f_{g+1}(k)}{f_{g+1}(k)} \ =:\ k'.$$

Hence, there are $r \le s$ such that $\{r, r + 1, \ldots, s\}$ is a subset of $\{1, \ldots, k\} \setminus P$ with a cardinality of at least k'. By the definition of P and the assumptions on $\pi, \pi_1, \ldots, \pi_k$, we have $\psi(\pi_r \ldots \pi_s) \in \{\angle_0, \ldots, \angle_g\}$. Let $g' \le g$ such that $\psi(\pi_r \ldots \pi_s) = \angle_{g'}$. We have

$$\Delta(\pi) \ \ge\ \Delta(\pi_r \ldots \pi_s) \ \ge\ f_{g'}(k') \ \ge\ f_g(k') \ =\ f_g\left(\frac{k - f_{g+1}(k)}{f_{g+1}(k)}\right) \ \ge\ f_g\left(\frac{k - f_{g+1}(k)}{f_{g+1}(k) + 1}\right)$$

which yields $f_{g+1}(k)$ by Lemma 3.14. □

Lemma 3.16. *Let $k \ge 1$ and $\pi_1, \ldots, \pi_k \in V^+$ such that $\psi(\pi_1), \ldots, \psi(\pi_k) \in \{\curlyvee_0, \ldots, \curlyvee_{h-1}\}$.*
We have $\Delta(\pi_1 \ldots \pi_k) \ \le\ 2 \max\{\Delta(\pi_1), \ldots, \Delta(\pi_k)\}$.

Proof. Let $0 \le g \le h$ be arbitrary. Let π' be some factor of $\pi_1 \ldots \pi_k$ with $\psi(\pi') = \angle_g$. We show $|\pi'|_g \ \le\ 2 \max\{\Delta(\pi_1), \ldots, \Delta(\pi_k)\}$.

Case 1: There is some $1 \le l \le k$ such that π' is a factor of π_l.
We have $|\pi'|_g \le \Delta(\pi')$ (by $\psi(\pi') = \angle_g$) and $\Delta(\pi') \le \Delta(\pi_l)$ by Lemma 3.12.

[2] We call some set $M \subseteq \{1, \ldots, k\} \setminus P$ a *maximal interval*, if there are $r \le s$ such that $M = \{r, r + 1, \ldots, s\}$ and $r-1 \notin M,\ s+1 \notin M$.

Case 2: There are $1 \le l < l' \le k$ such that $\pi' = \tilde{\pi}_l \pi_{l+1} \dots \pi_{l'-1} \tilde{\pi}_{l'}$, where $\tilde{\pi}_l$ (resp. $\tilde{\pi}_{l'}$) is a suffix of π_l (resp. prefix of $\pi_{l'}$).

By contradiction, assume that there is some $\angle_g$ in $\pi_{l+1} \dots \pi_{l'-1}$, i.e., $\angle_g \sqsubseteq \psi(\pi_{l+1} \dots \pi_{l'-1})$. However, $\psi(\pi_{l+1} \dots \pi_{l'-1}) \sqsubseteq \psi(\pi') = \angle_g$, i.e., $\psi(\pi_{l+1} \dots \pi_{l'-1}) = \angle_g$ which contradicts the assumption of the lemma. Hence, there is no $\angle_g$ in $\pi_{l+1} \dots \pi_{l'-1}$. Thus, $|\pi'|_g = |\tilde{\pi}_l|_g + |\tilde{\pi}_{l'}|_g$.

We show $|\tilde{\pi}_l|_g \le \Delta(\pi_l)$. We have $\psi(\tilde{\pi}_l) \sqsubseteq \psi(\pi') = \angle_g$. If $\psi(\tilde{\pi}_l) \sqsubset \angle_g$, then $|\tilde{\pi}_l|_g = 0$. If $\psi(\tilde{\pi}_l) = \angle_g$, then $|\tilde{\pi}_l|_g \le \Delta(\tilde{\pi}_l) \le \Delta(\pi_l)$.

Similarly, we obtain $|\tilde{\pi}_{l'}|_g \le \Delta(\pi_{l'})$. Thus, $|\pi'|_g \le \Delta(\pi_l) + \Delta(\pi_{l'})$.

□

Lemma 3.17. *Let $k \ge 1$, $\pi_1, \dots, \pi_{2k} \in V^*$, and $0 \le g < h$ such that for every $1 \le l \le k$:*

1. $\pi_{2l-1} = \varepsilon$ *or* $\psi(\pi_{2l-1}) \le \angle_g$ *and*
2. $\pi_{2l} \ne \varepsilon$ *and* $\psi(\pi_{2l}) \le \Upsilon_g$.

Then, we have $\Delta(\pi_1 \dots \pi_{2k}) \le 4 \max\{\Delta(\pi_1), \dots, \Delta(\pi_{2k})\}$.

Proof. For every $1 \le l \le k$, we have $\psi(\pi_{2l-1}\pi_{2l}) \in \{\Upsilon_0, \dots, \Upsilon_{h-1}\}$. Hence, we can apply Lemma 3.16 on $(\pi_1\pi_2), (\pi_3\pi_4), \dots, (\pi_{2k-1}\pi_{2k})$. □

Note that by Lemma 3.12, it follows that $\Delta(\pi_1 \dots \pi_{2k-1})$, $\Delta(\pi_2 \dots \pi_{2k})$, and $\Delta(\pi_2 \dots \pi_{2k-1})$ are at most $4 \max\{\Delta(\pi_1), \dots, \Delta(\pi_{2k})\}$.

3.5 Bibliographic Remarks

The notions in Section 3.1 belong to classic semigroup theory.

The definition of a consistent mapping was introduced by the author as an abstraction of certain transformations by I. Simon and H. Leung [29, 30, 31, 32, 34, 51, 52, 53]. The results in Section 3.2 are straightforward abstractions from similar results due to I. Simon and H. Leung which are shown in, e.g., [29, 30, 34]. Example 3.1 originates from the author.

The notions in Section 3.3 originate from the author. The nested distance desert semiring $\mathcal{V}$ is a joint generalization and further development of two semirings $\mathcal{R}$ resp. $\mathcal{D}$ which were introduced by H. Leung [29, 30] resp. the author [22, 25] as explained at the end of Section 3.3.

The results in Section 3.4 are due to the author.

4 The Decidability of Limitedness

In this section, we almost prove Theorem 2.1. Our solution is essentially a fusion and a further development of ideas from K. Hashiguchi, H. Leung, I. Simon, and the author [11, 15, 25, 29, 30, 34, 50, 51, 53]. We will prove (2)⇒(3) in Section 4.2 up to the bound on the $\sharp$-height of r which will be considered in Section 5.3. From Section 4.3 to 4.5, we develop some tools to prove (1)⇒(2) in Theorem 2.1 in Section 4.6.

For the entire Section 4, let $h \in \mathbb{N}$ and $\mathcal{A} = [Q, E, I, F, \theta]$ be an h-nested distance desert automaton. Let $n := |Q|$ and assume $Q = \{1, \ldots, n\}$. We denote by T the transformation matrices of letters, i.e., $T := \Psi(\Sigma)$. Clearly, $\langle T \rangle = \Psi(\Sigma^+)$.

4.1 Stabilization

We define a mapping $\sharp : \mathcal{V} \to \mathcal{V}$ which we call stabilization. For every $z \in \mathcal{V}$ let

$$z^\sharp := \begin{cases} z & \text{if } z \in \{\curlyvee_0, \ldots, \curlyvee_{h-1}\} \\ \omega & \text{if } z \in \{\angle_0, \ldots, \angle_h, \omega\} \\ \infty & \text{if } z = \infty \end{cases}$$

We have $z \leq z^\sharp$ for every $z \in \mathcal{V}$.

If $z \in \{\curlyvee_0, \ldots, \curlyvee_{h-1}, \omega, \infty\}$, then we have $z = z^\sharp$, and thus, $zz^\sharp = z^\sharp z^\sharp = z^\sharp$. If $z \in \{\angle_0, \ldots, \angle_h\}$, then $z^\sharp = \omega$, and consequently, $zz^\sharp = z^\sharp$. To sum up, we have $zz^\sharp = z^\sharp z = z^\sharp$ for every $z \in \mathcal{V}$.

We define $\sharp : \mathsf{E}(\mathcal{V}_{n\times n}) \to \mathcal{V}_{n\times n}$. For every $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ and i, j let

$$e^\sharp[i,j] := \mathsf{min}_{1\leq l\leq n} \left(e[i,l] \cdot (e[l,l])^\sharp \cdot e[l,j]\right).$$

This mapping is a joint generalization of I. Simon's and H. Leung's stabilization for idempotent matrices over $\mathcal{R}$ [29, 30, 34, 51, 53] and the author's stabilization for idempotent matrices over $\mathcal{D}$ [22, 25].

We show a remark to get familiar with stabilization.

Remark 4.1. Let $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ and i, j be arbitrary.

1. Let $0 \leq g \leq g' \leq h$, and assume $e[i,j] = \angle_g$ but $e[j,j] = \curlyvee_{g'}$. It is easy to see that $e[i,j] = e^2[i,j] \leq e[i,j] \cdot e[j,j] = \curlyvee_{g'}$, which is a contradiction. Hence, i, j with these properties cannot exist. Similarly, it is impossible that for some $0 \leq g < g' \leq h$, we have $e[i,j] = \curlyvee_g$ and $e[j,j] = \curlyvee_{g'}$.

2. We have $e^\sharp[i,j] \neq \angle_0$ by the definition of $e^\sharp$.

3. We have $e[i,j] = e^3[i,j] \leq e^\sharp[i,j]$.

4. Assume $e[i,j] \neq \infty$. By Lemma 3.11, there is some l such that $e[i,l] \cdot e[l,l] \cdot e[l,j] = e[i,j]$. Consequently, $\infty \neq e[i,l] \cdot (e[l,l])^\sharp \cdot e[l,j] \geq e^\sharp[i,j]$, i.e., $e^\sharp[i,j] \neq \infty$.

   Together with (3), we obtain $e \approx e^\sharp$.

5. If $e[i,j] = \omega$, then (3) and (4) imply $e^\sharp[i,j] = \omega$.

6. If $e[i,i] \in \{\curlyvee_0, \ldots, \curlyvee_{h-1}\}$, then $e^\sharp[i,i] \leq e[i,i] \cdot (e[i,i])^\sharp \cdot e[i,i] = e[i,i]$ by the definition of stabilization. In combination with (3), we obtain $e^\sharp[i,i] = e[i,i]$. □

For subsets $T \subseteq \mathcal{V}_{n\times n}$ we define $\langle T\rangle^\sharp$ as the least subset of $\mathcal{V}_{n\times n}$ which contains T and is closed both under matrix multiplication and stabilization $\sharp$ of idempotent matrices. It is easy to see that $\langle T\rangle^\sharp$ can be effectively computed.

4.2 On ♯-Expressions

Recall that we defined the notion of a ♯-expression already in Section 2.2.

We define a partial mapping τ from the set of ♯-expressions to $\mathcal{V}_{n\times n}$. The mapping τ extends Ψ to ♯-expressions. For every $a \in \Sigma$ let $\tau(a) := \Psi(a)$. If r and s are ♯-expressions and $\tau(r)$ and $\tau(s)$ are defined, then let $\tau(rs) := \tau(r)\tau(s)$. If r is a ♯-expression, $\tau(r)$ is defined, and $\tau(r) \in \mathsf{E}(\mathcal{V}_{n\times n})$, then let $\tau(r^\sharp) := \tau(r)^\sharp$.

If r is a ♯-expression and $\tau(r)$ is defined, then we call r a *typed ♯-expression* and we call $\tau(r)$ the type of r. For every typed ♯-expression r, we have $\tau(r) \in \langle T\rangle^\sharp$. Moreover, for every $a \in \langle T\rangle^\sharp$, there is a ♯-expression r such that $\tau(r) = a$.

Lemma 4.1. *Let r be a typed ♯-expression and $k \geq 1$.*

1. *Let i,j be arbitrary. There is some path $i \overset{r(k)}{\rightsquigarrow} j$ iff $\tau(r)[i,j] < \infty$.*
2. *We have $r(k) \in L(\mathcal{A})$ iff $I \cdot \tau(r) \cdot F < \infty$.*

Proof. We show (1). If r is a letter, then we have $\tau(r)[i,j] = \Psi(r)[i,j]$, and the claim is obvious.

Let r and s be typed ♯-expressions and assume by induction that (1) is true for r and s.

Assume that there is some path in $i \overset{rs(k)}{\rightsquigarrow} j$. Hence, there is some l such that there are paths in $i \overset{r(k)}{\rightsquigarrow} l$ and $l \overset{s(k)}{\rightsquigarrow} j$. Thus, $\tau(r)[i,l] \neq \infty$ and $\tau(s)[l,j] \neq \infty$. Consequently, $\tau(rs)[i,j] = \big(\tau(r)\tau(s)\big)[i,j] \leq \tau(r)[i,l] \cdot \tau(s)[l,j] < \infty$.

Conversely, assume $\tau(rs)[i,j] < \infty$. Hence, there is some l such that $\tau(r)[i,l] \cdot \tau(s)[l,j] < \infty$, i.e., $\tau(r)[i,l] < \infty$ and $\tau(s)[l,j] < \infty$. Thus, there are paths in $i \overset{r(k)}{\rightsquigarrow} l$ and $l \overset{s(k)}{\rightsquigarrow} j$. Since $r(k) \cdot s(k) = rs(k)$, there is some path in $i \overset{rs(k)}{\rightsquigarrow} j$.

Let r be a typed ♯-expression such that $\tau(r) \in \mathsf{E}(\mathcal{V}_{n\times n})$ and assume that r satisfies (1).

Assume that there is some path in $i \overset{r^\sharp(k)}{\rightsquigarrow} j$. Since $r^\sharp(k) = (r(k))^k$, there are $i = i_0, \ldots, i_k = j$ such that for every $1 \leq l \leq k$, there is some path $i_{l-1} \overset{r(k)}{\rightsquigarrow} i_l$. Hence, for every $1 \leq l \leq k$, we have $\tau(r)[i_{l-1}, i_l] < \infty$. Thus, $\tau(r)^k[i,j] < \infty$, i.e., $\tau(r)[i,j] < \infty$, and by Remark 4.1(4), we have $\tau(r^\sharp)[i,j] = \tau(r)^\sharp[i,j] < \infty$.

Conversely, assume $\tau(r)[i,j] < \infty$. Since $\tau(r)[i,j] = \tau(r)^k[i,j]$ there are $i = i_0, \ldots, i_k = j$ such that for every $1 \leq l \leq k$, we have $\tau(r)[i_{l-1}, i_l] < \infty$, and hence, by Remark 4.1(1), $\tau(r^\sharp)[i_{l-1}, i_l] < \infty$. Thus, for every $1 \leq l \leq k$, there is some path $i_{l-1} \overset{r(k)}{\rightsquigarrow} i_l$. Consequently, there is some path $i \overset{r^\sharp(k)}{\rightsquigarrow} j$.

Assertion (2) is an immediate consequence of (1). □

Proposition 4.2. *Let r be a typed ♯-expression.*
For every bound $d \geq 0$, there is some $K \geq 1$ such that for every $k \geq K$, we have:
For every i,j and every path $\pi \in i \overset{r(k)}{\rightsquigarrow} j$ such that $\psi(\theta(\pi)) < \tau(r)[i,j]$, we have $\Delta(\theta(\pi)) \geq d$.

Proof. We proceed by an induction on typed ♯-expressions.

If r is just a letter, then $\tau(r)[i,j] = \Psi(r)[i,j]$. Paths $\pi \in i \overset{r}{\rightsquigarrow} j$ such that $\psi(\theta(\pi)) < \tau(r)[i,j] = \Psi(r)[i,j]$ cannot exist, and we are done.

Let r and s be typed $\sharp$-expressions and assume that the claim is true for r and s. Let $d \geq 0$ be arbitrary, and let K be the maximum of the corresponding integers K for r and s, and let $k \geq K$.

Let i, j be arbitrary, and let $\pi \in i \overset{rs(k)}{\rightsquigarrow} j$ such that $\psi(\theta(\pi)) < \tau(rs)[i,j]$. There is some l, $\pi_1 \in i \overset{r(k)}{\rightsquigarrow} l$, and $\pi_2 \in l \overset{s(k)}{\rightsquigarrow} j$ such that $\pi = \pi_1\pi_2$.

If $\psi(\theta(\pi_1)) < \tau(r)[i,l]$, then we have by induction $\Delta(\pi_1) \geq d$, i.e., $\Delta(\pi_1\pi_2) \geq d$. If $\psi(\theta(\pi_2)) < \tau(s)[l,j]$, then we have $\Delta(\pi_1\pi_2) \geq d$ in the same way. It remains to consider the case that $\psi(\theta(\pi_1)) \geq \tau(r)[i,l]$ and $\psi(\theta(\pi_2)) \geq \tau(s)[l,j]$. We obtain

$$\psi(\theta(\pi)) \;=\; \psi(\theta(\pi_1))\psi(\theta(\pi_2)) \;\geq\; \tau(r)[i,l] \cdot \tau(s)[l,j] \;\geq\; \tau(rs)[i,j],$$

and we are done.

Finally, let r be a typed $\sharp$-expression such that $\tau(r) \in \mathsf{E}(\mathcal{V}_{n\times n})$ and assume that the claim is true for r. We show the claim for $r^\sharp$. Let $e := \tau(r)$. Let $d \geq 0$ be arbitrary, and let K be an integer which satisfies the condition

$$\sqrt[h+1]{\frac{K-1}{n}} - 1 \;\geq\; d.$$

Moreover, we assume that K is not smaller than the corresponding integer for r. Let $k \geq K$ and $\pi \in i \overset{r^\sharp(k)}{\rightsquigarrow} j$. There are $i = i_0, \dots, i_k = j$ and for every $1 \leq p \leq k$, some $\pi_p \in i_{p-1} \overset{r(k)}{\rightsquigarrow} i_p$ such that $\pi = \pi_1 \dots \pi_k$.

Let $1 \leq p \leq k$ be arbitrary. If $\psi(\theta(\pi_p)) < e[i_{p-1}, i_p]$, then we have by the inductive hypothesis $\Delta(\pi_p) \geq d$, and thus, $\Delta(\pi) \geq d$, and we are done. Hence, we assume $\psi(\theta(\pi_p)) \geq \tau(r)[i_{p-1}, i_p]$ in the rest of the proof.

Let $1 \leq p < q \leq k$ be arbitrary. We have

$$\psi(\theta(\pi_{p+1} \dots \pi_q)) = \psi(\theta(\pi_{p+1}) \cdots \psi(\theta(\pi_q)) \;\geq\; e[i_p, i_{p+1}] \cdots e[i_{q-1}, i_q] \;\geq\; e^{q-p}[i_p, i_q] \;=\; e[i_p, i_q]. \quad (2)$$

There is some l such that the set $I := \{p \,|\, 1 \leq p < k,\; i_p = l\}$ contains at least $\frac{k-1}{n}$ members.

Case 1: There are $p < q \in I$ such that $\psi(\theta(\pi_{p+1} \dots \pi_q)) \in \{\Upsilon_0, \dots, \Upsilon_{h-1}\}$.

By 2, we have $e[l,l] = e[i_p, i_q] \leq \psi(\theta(\pi_{p+1} \dots \pi_q))$, i.e., $e[l,l] \in \{\Upsilon_0, \dots, \Upsilon_{h-1}\}$, and in particular, $(e[l,l])^\sharp = e[l,l]$. In the same way, we obtain $e[i,l] \leq \psi(\theta(\pi_1 \dots \pi_p))$ and $e[l,j] \leq \psi(\theta(\pi_{q+1} \dots \pi_k))$. To sum up,

$$\psi(\theta(\pi)) = \psi(\theta(\pi_1 \dots \pi_k)) \;\geq\; e[i,l] \cdot e[l,l] \cdot e[l,j] \;=\; e[i,l] \cdot (e[l,l])^\sharp \cdot e[l,j] \;\geq\; e^\sharp[i,j].$$

Hence, $\psi(\theta(\pi)) \geq \tau(r^\sharp)[i,j]$, and we are done.

Case 2: For every $p < q \in I$, we have $\psi(\theta(\pi_{p+1} \dots \pi_q)) \in \{\angle_0, \dots, \angle_h\}$.

There are $|I| - 1$ consecutive factors in π on which the image under $\psi \circ \theta$ belongs to $\{\angle_0, \dots, \angle_h\}$. By Lemma 3.15, we have $\Delta(\pi) \geq \sqrt[h+1]{|I|} - 1$. By $|I| \geq \frac{k-1}{n}$ and $k \geq K$, we have $|I| \geq \frac{K-1}{n}$. By the choice of K, we obtain $\Delta(\pi) \geq \sqrt[h+1]{\frac{K-1}{n}} - 1 \geq d$.

□

Proposition 4.3. *Let r be a typed $\sharp$-expression such that $I \cdot \tau(r) \cdot F = \omega$. For every $k \geq 1$, we have $r(k) \in L(\mathcal{A})$, and $\Delta_{\mathcal{A}}\big(r(k)\big)$ is unbounded for increasing integers k.*

Proof. Let r be a typed $\sharp$-expression such that $I \cdot \tau(r) \cdot F = \omega$. For every $k \geq 1$, we have $r(k) \in L(\mathcal{A})$ by Lemma 4.1(2).

Let $d \geq 0$ be arbitrary, and let K be the integer provided by Proposition 4.2. To prove the assertion, we show that for every $k \geq K$, we have $\Delta(r(k)) \geq d$. Let $k \geq K$ and let π be a successful path for $r(k)$. Let $i \in I$ and $j \in F$ be the first (resp. last state of π). By Lemma 4.1(1), we have $\tau(r)[i,j] \neq \infty$, and since $I \cdot \tau(r) \cdot F = \omega$, we have $\tau(r)[i,j] \geq \omega$, i.e., $\tau(r)[i,j] = \omega$. Moreover, $\psi(\theta(\pi)) < \omega = \tau(r)[i,j]$. By Proposition 4.2, we have $\Delta(\theta(\pi)) \geq d$. From the arbitrary choice of π, it follows $\Delta(r(k)) \geq d$. ☐

Proposition 4.3 almost proves (2)⇒(3) in Theorem 2.1. However, we have to invest some more ideas to construct a desired $\sharp$-expression r which is of $\sharp$-height of at most $(h+1)n$.

4.3 Stabilization is a Consistent Mapping

The aim of this section is to show that stabilization is a consistent mapping. At first, we show a preliminary lemma:

Lemma 4.4. *Let $e \in \mathsf{E}(\mathcal{V}_{n\times n})$. We have $e^\sharp = ee^\sharp = e^\sharp e = ee^\sharp e = e^\sharp e^\sharp$.*

Proof. Let i, j be arbitrary.

At first, we show $(ee^\sharp)[i,j] \geq e^\sharp[i,j]$. Let k such that $(ee^\sharp)[i,j] = e[i,k] \cdot e^\sharp[k,j]$, and let l such that $e^\sharp[k,j] = e[k,l] \cdot (e[l,l])^\sharp \cdot e[l,j]$. We obtain

$$(ee^\sharp)[i,j] \;=\; e[i,k] \cdot e[k,l] \cdot (e[l,l])^\sharp \cdot e[l,j] \;\geq\; e[i,l] \cdot (e[l,l])^\sharp \cdot e[l,j] \;\geq\; e^\sharp[i,j].$$

Now, we show $e^\sharp[i,j] \geq (ee^\sharp)[i,j]$. Let l such that $e^\sharp[i,j] = e[i,l] \cdot (e[l,l])^\sharp \cdot e[l,j]$, and let k such that $e[i,l] = e[i,k] \cdot e[k,l]$. We obtain

$$e^\sharp[i,j] \;=\; e[i,k] \cdot e[k,l] \cdot (e[l,l])^\sharp \cdot e[l,j] \;\geq\; e[i,k] \cdot e^\sharp[k,j] \;\geq\; (ee^\sharp)[i,j].$$

To sum up, $e^\sharp[i,j] = (ee^\sharp)[i,j]$, i.e., $e^\sharp = ee^\sharp$. We can show $e^\sharp = e^\sharp e$ in a symmetric way, and from $e^\sharp = ee^\sharp = e^\sharp e$, we obtain immediately $e^\sharp = ee^\sharp e$.

It remains to show $e^\sharp = e^\sharp e^\sharp$. By Remark 4.1(3), we have $e^\sharp \geq e$, and hence, $e^\sharp e^\sharp \geq ee^\sharp = e^\sharp$. Let i, j be arbitrary. We show $(e^\sharp e^\sharp)[i,j] \leq e^\sharp[i,j]$. Let l such that $e^\sharp[i,j] = e[i,l] \cdot (e[l,l])^\sharp \cdot e[l,j]$. Since for every $z \in \mathcal{V}$, we have $z^\sharp = z^\sharp z^\sharp = z^\sharp z z z^\sharp$, we obtain

$$e^\sharp[i,j] \;=\; e[i,l] \cdot (e[l,l])^\sharp \cdot e[l,j] \;=\; e[i,l] \cdot (e[l,l])^\sharp \cdot e[l,l]) \cdot e[l,l] \cdot (e[l,l])^\sharp \cdot e[l,j] \;\geq\; \ldots$$

$$\ldots \geq e^\sharp[i,l] \cdot e^\sharp[l,j] \;\geq\; (e^\sharp e^\sharp)[i,j].$$

To sum up, $e^\sharp \geq e^\sharp e^\sharp$. ☐

Proposition 4.5. *Stabilization $\sharp$ on $\mathcal{V}_{n\times n}$ is a consistent mapping.*

Proof. By Lemma 4.4, we have $e^\sharp \in \mathsf{E}(\mathcal{V}_{n\times n})$ for every $e \in \mathsf{E}(\mathcal{V}_{n\times n})$. Hence, $\sharp$ is indeed a mapping from $\mathsf{E}(\mathcal{V}_{n\times n})$ to $\mathsf{E}(\mathcal{V}_{n\times n})$.

Let $a, b \in \mathcal{V}_{n\times n}$ and let $e, f \in \mathsf{E}(\mathcal{V}_{n\times n})$ such that $e =_{\mathscr{J}} f$, and in particular, $f = aeb$. To show that $\sharp$ is a consistent mapping, we have to show $f^\sharp = ae^\sharp b$.

We have $f = (ae)(eb)$ and $(ae) =_{\mathscr{J}} (eb) =_{\mathscr{J}} e$. We show $e = ebae$. We denote GREEN's relations between e, f, ae, and eb in the following egg-box picture:

eb		e	
f		ae	

Because the idempotent f belongs to $\mathscr{L}(eb) \cap \mathscr{R}(ae)$, we have by Lemma 3.4, $(eb)(ae) =_{\mathscr{J}} e$. Thus, $ebae =_{\mathscr{L}} e$, and $ebae =_{\mathscr{R}} e$, i.e., $ebae =_{\mathscr{H}} e$. Moreover, we have $(ebae)(ebae) = ebfae = ebae \in \mathsf{E}(S)$. By Lemma 3.3, $\mathscr{H}(e)$ is a group. Thus, there is exactly one idempotent in $\mathscr{H}(e)$, and hence, $ebae = e$.

We show $f^\sharp \leq ae^\sharp b$. Let i, j be arbitrary.

Let r, s such that $aee^\sharp eb[i,j] = (ae)[i,r] \cdot e^\sharp[r,s] \cdot (eb)[s,j]$.

Let l such that $e^\sharp[r,s] = e[r,l] \cdot \big(e[l,l]\big)^\sharp \cdot e[l,s]$.

Let l' such that $e[l,l] = (eb)[l,l'] \cdot (ae)[l',l]$. By setting $x = (eb)[l,l']$ and $y = (ae)[l',l]$, we obtain

$$\big(e[l,l]\big)^\sharp \;=\; (xy)^\sharp \;=\; (xy)^\sharp xy \;=\; x(yx)^\sharp y \;=\; (eb)[l,l'] \cdot \Big((ae)[l',l] \cdot (eb)[l,l']\Big)^\sharp \cdot (ae)[l',l].$$

We have

$$aee^\sharp eb[i,j] \;=\; (ae)[i,r] \cdot \underbrace{e[r,l] \cdot \big(e[l,l]\big)^\sharp \cdot e[l,s]}_{=e^\sharp[r,s]} \cdot (eb)[s,j] \;=\; \ldots$$

$$\ldots = (ae)[i,r] \cdot e[r,l] \cdot \underbrace{(eb)[l,l'] \cdot \Big((ae)[l',l] \cdot (eb)[l,l']\Big)^\sharp \cdot (ae)[l',l]}_{=(e[l,l])^\sharp} \cdot e[l,s] \cdot (eb)[s,j] \;\geq\; \ldots$$

$$\ldots \;\geq\; (aeb)[i,l'] \cdot \Big((aeb)[l',l']\Big)^\sharp \cdot (aeb)[l',j] \;=\; f[i,l'] \cdot (f[l',l'])^\sharp \cdot f[l',j] \;\geq\; f^\sharp[i,j].$$

Hence, $(aee^\sharp eb)[i,j] \geq f^\sharp[i,j]$. By Lemma 4.4, we have $(ae^\sharp b)[i,j] \geq f^\sharp[i,j]$, i.e., $(ae^\sharp b) \geq f^\sharp$.

We have seen $ebfae = e$. As above, we can show $ebf^\sharp ae \geq e^\sharp$. Hence, we have $aebf^\sharp aeb \geq ae^\sharp b$, i.e., $ff^\sharp f \geq ae^\sharp b$, and by Lemma 4.4, $f^\sharp \geq ae^\sharp b$.

To sum up, $f^\sharp = ae^\sharp b$. □

By Lemma 4.5 and Corollary 3.9 we have a natural extension of stabilization to $\mathsf{Reg}(\mathcal{V}_{n\times n})$, and we can use Lemma 3.10 as a very convenient tool whenever we prove some assertion concerning stabilization.

At this point, we have to be very careful with the definition of $\langle T\rangle^\sharp$. Let $a \in \mathsf{Reg}(\langle T\rangle^\sharp)$. There is some $e \in \mathsf{E}(\langle T\rangle^\sharp)$ with $e =_{\mathscr{L}} a$. Then, $ae = a$ and $a^\sharp = ae^\sharp$, and thus, $a^\sharp \in \langle T\rangle^\sharp$, or more precisely, $a^\sharp \in \mathsf{Reg}(\langle T\rangle^\sharp)$. Consequently, $\langle T\rangle^\sharp$ is closed under stabilization of matrices in $\mathsf{Reg}(\langle T\rangle^\sharp)$.

However, for $b \in \mathsf{Reg}(\mathcal{V}_{n\times n})$, it is possible that $b \notin \mathsf{Reg}(\langle T\rangle^\sharp)$ and $b^\sharp \notin \langle T\rangle^\sharp$.

In the definition of $\langle T\rangle^\sharp$ we demand closure under stabilization of idempotents. *After* the definition of $\langle T\rangle^\sharp$ is given, we proved closure under stabilization of matrices which are regular in $\langle T\rangle^\sharp$.

If one defines $\langle T\rangle^\sharp$ in a way that $\langle T\rangle^\sharp$ has to be closed under stabilization of matrices which are regular in $\langle T\rangle^\sharp$, then the definition becomes a mess, because the term "regular matrix in $\langle T\rangle^\sharp$" does not have a meaning unless $\langle T\rangle^\sharp$ is defined.

4.4 Stabilization of Regular Matrices

We show two crucial lemmas about stabilization of regular matrices in $\mathcal{V}_{n\times n}$. Since $\mathsf{Reg}(\langle T\rangle^\sharp) \subseteq \mathsf{Reg}(\mathcal{V}_{n\times n})$, we can apply both lemmas for matrices in $\mathsf{Reg}(\langle T\rangle^\sharp)$.

Lemma 4.6. *For every* $a \in \mathsf{Reg}(\mathcal{V}_{n\times n})$, *we have*

1. $a \leq a^\sharp$, $a \approx a^\sharp$, *and*
2. *for every* i, j, $a^\sharp[i,j] \neq \angle_0$.

Proof. Let $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ with $e =_{\mathscr{L}} a$, i.e., $a = ae$, and $a^\sharp = ae^\sharp$. (1) is an immediate conclusion from Remark 4.1(3)(4), and the stability of $\leq$ and $\approx$ under matrix multiplication.

We have (2), because $\angle_0$ cannot occur in $e^\sharp$ by Remark 4.1(2). □

Lemma 4.7. *Let* $a, b, c \in \mathcal{V}_{n\times n}$ *be a smooth product in* $\mathcal{V}_{n\times n}$ *and let* i, j *such that we have* $(abc)^\sharp[i,j] \in \{\angle_1, \dots, \angle_h\}$. *Then, there are* p, q *such that*

1. $a[i,p] \cdot b^\sharp[p,q] \cdot c[q,j] \;=\; (abc)^\sharp[i,j]$,
2. $b^\sharp[p,q] \;<\; (abc)^\sharp[i,j]$, *and* $b^\sharp[p,q] \;\sqsubset\; (abc)^\sharp[i,j]$.

The reader should be aware that we state and prove Lemma 4.7 for smooth products in $\mathcal{V}_{n\times n}$, i.e., for $a, b, c \in \mathcal{V}_{n\times n}$ satisfying $a =_{\mathscr{J}} b =_{\mathscr{J}} c =_{\mathscr{J}} abc$ in $\mathcal{V}_{n\times n}$.

Now, let $a, b, c \in \langle T\rangle^\sharp$ and assume that $a =_{\mathscr{J}} b =_{\mathscr{J}} c =_{\mathscr{J}} abc$ holds in $\langle T\rangle^\sharp$. Hence, we have $a, b, c, abc \in \mathsf{Reg}(\langle T\rangle^\sharp) \subseteq \mathsf{Reg}(\mathcal{V}_{n\times n})$ and it holds $a =_{\mathscr{J}} b =_{\mathscr{J}} c =_{\mathscr{J}} abc$ holds in $\mathcal{V}_{n\times n}$. Consequently, a, b, c satisfy the assumptions of Lemma 4.7, i.e., we can apply Lemma 4.7 on a, b, c and we have $(abc)^\sharp \in \mathsf{Reg}(\langle T\rangle^\sharp)$.

Proof of Lemma 4.7. Let $0 \leq g \leq h$ such that $(abc)^\sharp[i,j] = \angle_g$. We denote GREEN's relations between a, b, c and their products in the following egg-box picture:

a		ab		abc
e		b		bc
d		f		c

By $a =_{\mathscr{J}} b =_{\mathscr{J}} ab$, there is an idempotent $e \in \mathscr{L}(a) \cap \mathscr{R}(b)$, and similarly, there is an idempotent $f \in \mathscr{L}(b) \cap \mathscr{R}(c)$. By Lemma 3.7, there is a $d \in \mathscr{L}(e) \cap \mathscr{R}(f)$ such that $bd = e$ and $db = f$.

We have $a = ae$, $abc = aebc$, and $(abc)^\sharp = ae^\sharp bc$.

Because $(abc)^\sharp[i,j] = ae^\sharp bc[i,j] = \angle_g$, there are r, s such that $a[i,r] \cdot e^\sharp[r,s] \cdot (bc)[s,j] = \angle_g$, and in particular $e^\sharp[r,s] \sqsubseteq \angle_g$. By the definition of stabilization, there is some p such that we have $e[r,p] \cdot (e[p,p])^\sharp \cdot e[p,s] \;=\; e^\sharp[r,s] \;\sqsubseteq\; \angle_g$, and hence, $(e[p,p])^\sharp \sqsubseteq \angle_g$. In combination with $(e[p,p])^\sharp \in \{\Upsilon_0, \dots, \Upsilon_{h-1}, \omega, \infty\}$, we obtain $(e[p,p])^\sharp \in \{\Upsilon_0, \dots, \Upsilon_{g-1}\}$, i.e., $e[p,p] \in \{\Upsilon_0, \dots, \Upsilon_{g-1}\}$. By Remark 4.1(6), we get $e^\sharp[p,p] = e[p,p]$. By $(e[p,p])^\sharp \in \{\Upsilon_0, \dots, \Upsilon_{g-1}\}$, we have in particular $e^\sharp[p,p] = e[p,p] \sqsubset \angle_g$.

We have $(bc)[s,j] \sqsubseteq \angle_g$ and $e[p,s] \sqsubseteq \angle_g$, and by Remark 3.3, $(bc)[s,j] \leq \angle_g$ and $e[p,s] \leq \angle_g$.

By $e^\sharp = b^\sharp d$, there is some q such that $b^\sharp[p,q] \cdot d[q,p] \;=\; e^\sharp[p,p] \;\sqsubset\; \angle_g$, i.e., $b^\sharp[p,q] \sqsubset \angle_g$ and $d[q,p] \sqsubset \angle_g$. By Remark 3.3, we have $b^\sharp[p,q] < \angle_g$ and $d[q,p] < \angle_g$ which proves (2).

We have $a[i,r] \sqsubseteq \angle_g$ and $e[r,p] \sqsubseteq \angle_g$, and thus, $a[i,r] \cdot e[r,p] \sqsubseteq \angle_g$, and by Remark 3.3, $a[i,r] \cdot e[r,p] \leq \angle_g$. Thus, $ae[i,p] = a[i,p] \leq \angle_g$.

We have $c = ffc = dbdbc = debc$, i.e., $c[q,j] = (debc)[q,j] \leq d[q,p] \cdot e[p,s] \cdot (bc)[s,j] \leq \angle_g \cdot \angle_g \cdot \angle_g = \angle_g$, i.e, $c[q,j] \leq \angle_g$.

Thus, $a[i,p] \cdot b^\sharp[p,q] \cdot c[q,j] \leq \angle_g \cdot \angle_g \cdot \angle_g \leq \angle_g$. On the other hand, $a[i,p] \cdot b^\sharp[p,q] \cdot c[q,j] \geq (ab^\sharp c)[i,j] = (abc)^\sharp[i,j] = \angle_g$. Consequently, $a[i,p] \cdot b^\sharp[p,q] \cdot c[q,j] = \angle_g$ which proves (1). ☐

A generalization of Lemma 4.7 to the case $(abc)^\sharp[i,j] = \angle_0$ is vacuously true, because we have $(abc)^\sharp[i,j] \neq \angle_0$ by Lemma 4.6(2). A generalization for $(abc)^\sharp[i,j] \in \{\Upsilon_0, \ldots, \Upsilon_{h-1}\}$ is not possible, just let $a = b = c$ be the matrix in which every entry is Υ_0.

4.5 On the Growth of Entries

We consider pairs in $\mathcal{V}_{n\times n} \times \Sigma^+$. For every pair $(a,w) \in \mathcal{V}_{n\times n} \times \Sigma^+$, let $\Delta'(a,w)$ be the least non-negative integer such that for every i,j with $a[i,j] \notin \{\omega, \infty\}$ there is some $\pi \in i \overset{w}{\leadsto} j$ such that $\psi(\theta(\pi)) \leq a[i,j]$ and $\Delta(\theta(\pi)) \leq \Delta'(a,w)$. If such an integer does not exist, then we set $\Delta'(a,w) := \infty$. More precisely, we set

$$\Delta'(a,w) \;:=\; \mathsf{max}_{i,j,\,a[i,j]\notin\{\omega,\infty\}}\, \mathsf{min}\Big\{\Delta(\theta(\pi)) \,\Big|\, \pi \in i \overset{w}{\leadsto} j,\ \psi(\theta(\pi)) \leq a[i,j]\Big\}.$$

The cartesian product $\mathcal{V}_{n\times n} \times \Sigma^+$ is a semigroup in a natural way whereas the operation is componentwise multiplication in $\mathcal{V}_{n\times n}$ and concatenation of words.

Proposition 4.8. *Let $k \geq 1$ and $(a_1, w_1), \ldots, (a_k, w_k) \in \mathcal{V}_{n\times n} \times \Sigma^+$.*

1. *We have $\Delta'\big(a_1 \ldots a_k,\ w_1 \ldots w_k\big) \;\leq\; k \cdot \mathsf{max}_{1\leq l\leq k} \Delta'(a_l, w_l)$.*

2. *If $a_1, \ldots, a_k$ are a smooth product in $\mathcal{V}_{n\times n}$, then*

$$\Delta'\big((a_1 \ldots a_k)^\sharp,\ w_1 \ldots w_k\big) \;\leq\; 2^{3h-1} \cdot \mathsf{max}_{1\leq l\leq k} \Delta'(a_l, w_l).$$

The most important fact in Proposition 4.8 is that the bound 2^{3h-1} in (2) does not depend on k. Although the bound 2^{3h-1} seems to be very large, this bound holds in contrast to (1) for arbitrarily large k.

Proof of Proposition 4.8. We denote $d := \mathsf{max}_{1\leq l\leq k} \Delta'(a_l, w_l)$, $a := a_1 \cdots a_k$, and $w := w_1 \ldots w_k$.

(1) Let i,j such that $a[i,j] \notin \{\omega, \infty\}$. There are $i = i_0, \ldots, i_k = j$ such that we have $a[i,j] = a_1[i_0, i_1] \cdots a_k[i_{k-1}, i_k]$. For every $1 \leq l \leq k$, there is some $\pi_l \in i_{l-1} \overset{w_l}{\leadsto} i_l$ such that $\psi(\theta(\pi_l)) \leq a_l[i_{l-1}, i_l]$ and $\Delta(\theta(\pi_l)) \leq d$. We have $\pi_1 \ldots \pi_k \in i \overset{w}{\leadsto} j$, $\Delta(\theta(\pi_1 \ldots \pi_k)) \leq kd$, and

$$\psi(\theta(\pi_1 \ldots \pi_k)) = \psi(\theta(\pi_1)) \cdots \psi(\theta(\pi_k)) \;\leq\; a_1[i_0, i_1] \cdots a_k[i_{k-1}, i_k] \;=\; a[i,j],$$

and (1) follows.

To show (2), we show the following two claims (a) and (b). Let i, j, and $0 \leq g \leq h$ be arbitrary.

(a) If $a^\sharp[i,j] = \angle_g$, then there is a $\pi \in i \overset{w}{\leadsto} j$ with $\psi(\theta(\pi)) \leq \angle_g$ and $\Delta(\theta(\pi)) \leq 2^{3g-1}d$.

(b) If $a^\sharp[i,j] = \Upsilon_g$, then there is a $\pi \in i \overset{w}{\leadsto} j$ with $\psi(\theta(\pi)) \leq \Upsilon_g$ and $\Delta(\theta(\pi)) \leq 2^{3g+1}d$.

We show both claims by an induction over the ordering $\sqsubseteq$ of $\mathcal{V}$.

For $\angle_0$, claim (a) is vacuously true, because $a^\sharp[i,j] \neq \angle_0$ by Lemma 4.6(2).

For $k = 1$, both claims (a) and (b) are obvious. For $k = 2$, both claims (a) and (b) follow from assertion (1) since $a \leq a^\sharp$ and $2 = k \leq 2^{3g-1}$ for $g \geq 1$ (resp. $2 = k \leq 2^{3g+1}$ for $g \geq 0$). However, for $k = 3$ we cannot simply use assertion (1) since for $g = 0$ in (b) we have $k = 3 \not\leq 2^{3\cdot 0+1} = 2$.

In the rest of the proof we assume $k \geq 3$.

We show (b) for Υ_0. By Lemma 3.10(2), $(a_1 \dots a_k)^\sharp = a_1^\sharp \dots a_k^\sharp$. Thus, $(a_1^\sharp \dots a_k^\sharp)[i,j] = \Upsilon_0$, i.e., there are $i = i_0, \dots, i_k = j$ such that for every $1 \leq l \leq k$, we have $a_l^\sharp[i_{l-1}, i_l] \in \{\Upsilon_0, \angle_0\}$, and by Lemma 4.6(2), $a_l^\sharp[i_{l-1}, i_l] = \Upsilon_0$.

Let $1 \leq l \leq k$ be arbitrary. By Lemma 4.6(1), we have $a_l[i_{l-1}, i_l] \in \{\Upsilon_0, \dots, \Upsilon_{h-1}\}$. Let $\pi_l \in i_{l-1} \overset{w_l}{\rightsquigarrow} i_l$ with $\psi(\theta(\pi_l)) \leq a_l[i_{l-1}, i_l]$ and $\Delta(\theta(\pi_l)) \leq d$. Then, $\psi(\theta(\pi_1 \dots \pi_k)) \in \{\Upsilon_0, \dots, \Upsilon_{h-1}\}$, i.e., $\psi(\theta(\pi_1 \dots \pi_k)) \leq \Upsilon_0$. By Lemma 3.16, we have $\Delta(\theta(\pi_1 \dots \pi_k)) \leq 2d$. Hence, $\pi := \pi_1 \dots \pi_k$ proves the claim.

Next, we show (a) for some $1 \leq g \leq h$, i.e., we assume $(a_1 \dots a_k)^\sharp[i,j] = \angle_g$. By induction, we assume that both (a) and (b) are true for $0 \leq g' < g$.

Since $k \geq 3$, we can apply Lemma 4.7 on $a_1(a_2 \dots a_{k-1})a_k$. Let p, q be from Lemma 4.7. Let $\pi_1 \in i \overset{w_1}{\rightsquigarrow} p$ with $\psi(\theta(\pi_1)) \leq a_1[i,p]$ and $\Delta(\theta(\pi_1)) \leq d$. Similarly, let $\pi_k \in q \overset{w_k}{\rightsquigarrow} j$ with $\psi(\theta(\pi_{|w|})) \leq a_k[p,j]$ and $\Delta(\theta(\pi_k)) \leq d$.

Let $z := (a_2 \dots a_{k-1})^\sharp[p,q]$. By Lemma 4.7(2), we have $z \sqsubset (a_1 \dots a_k)^\sharp[i,j] = \angle_g$, i.e., we can apply the inductive hypothesis on $(a_2 \dots a_{k-1})[p,q]$. Thus, there is a $\tilde{\pi} \in p \overset{w_2 \dots w_{k-1}}{\rightsquigarrow} q$ such that $\psi(\theta(\tilde{\pi})) \leq (a_2 \dots a_{k-1})^\sharp[p,q]$ and $\Delta(\theta(\tilde{\pi})) \leq 2^{3g-2}d$. We have by Lemma 4.7(1)

$$\psi(\theta(\pi_1)) \cdot \psi(\theta(\tilde{\pi})) \cdot \psi(\theta(\pi_k)) \;\leq\; a_1[i,p] \cdot (a_2 \dots a_{k-1})^\sharp[p,q] \cdot a_k[q,j] \;=\; a^\sharp[i,j].$$

Moreover, we have $\Delta(\theta(\pi_1\tilde{\pi}\pi_k)) \leq 2d + 2^{3g-2}d \leq (2 + 2^{3g-2})d \leq 2^{3g-1}d$, i.e., claim (a) is true for $\pi := \pi_1\tilde{\pi}\pi_k$. The estimation $(2 + 2^{3g-2}) \leq 2^{3g-1}$ is rough, but we rather want to avoid technical overhead.

At last, we show claim (b) for some $1 \leq g < h$, i.e., we assume $(a_1 \dots a_k)^\sharp[i,j] = \Upsilon_g$. By induction, we assume that (a) and (b) are true for $\angle_0, \dots, \angle_g$ (resp. $\Upsilon_0, \dots, \Upsilon_{g-1}$).

By Lemma 3.10(2), we have $(a_1 \dots a_k)^\sharp[i,j] = (a_1^\sharp \dots a_k^\sharp)[i,j] = \Upsilon_g$. There is at least one sequence $i = i_0, \dots, i_k = j$, such that

$$a_1^\sharp[i_0, i_1] \cdots a_k^\sharp[i_{k-1}, i_k] \;=\; \Upsilon_g.$$

We choose some sequence $i_0, \dots, i_k$ with this property such that $a_l^\sharp[i_{l-1}, i_l] = \Upsilon_g$ for as many $1 \leq l \leq k$ as possible. Note that there is at least one l with $a_l^\sharp[i_{l-1}, i_l] = \Upsilon_g$.

Now, let $1 \leq r \leq s \leq k$ such that $z_2 := a_r^\sharp[i_{r-1}, i_r] \dots a_s^\sharp[i_{s-1}, i_s] \sqsubset \Upsilon_g$. Moreover, we assume that either $r = 1$ or $a_{r-1}^\sharp[i_{r-2}, i_{r-1}] = \Upsilon_g$. Similarly, we assume that $s = k$ or $a_{s+1}^\sharp[i_s, i_{s+1}] = \Upsilon_g$. Let us mention that it is not clear whether such r and s exist. However, the existence of r, s is not really important, we just want to develop some arguments which are required *if* there are r, s with these properties.

Let $z := (a_r^\sharp \dots a_s^\sharp)[i_{r-1}, i_s]$. We derive information on z.

- We have $z \leq z_2 \sqsubset \Upsilon_g$. Hence, $z_2 \sqsubseteq \angle_g$, and by Remark 3.3, $z_2 \leq \angle_g$. Thus, we have $z \leq \angle_g$.
- By contradiction, assume $z = \Upsilon_g$. If $r = s$, then we have $z_2 = \Upsilon_g$ which contradicts the choice of r and s. If $r < s$, then we can replace in $i_0, \dots, i_k$ the indices $i_r, \dots, i_{s-1}$ by $i'_r, \dots, i'_{s-1}$ such that $a_r^\sharp[i_{r-1}, i'_r] \cdots a_s^\sharp[i'_{s-1}, i_s] = \Upsilon_g$. Then, we have a contradiction to the choice of $i_0, \dots, i_{|w|}$, or more precisely, to the condition that $a_l^\sharp[i_{l-1}, i_l] = \Upsilon_g$ for as many l as possible.

- By contradiction, assume $z \in \{\Upsilon_{g+1}, \dots, \Upsilon_{h-1}\}$. We conclude

$$\Upsilon_g \;=\; (a_1^\sharp \dots a_k^\sharp)[i,j] \;\leq\; \dots$$

$$\underbrace{a_1^\sharp[i_0,i_1]\cdots a_{r-1}^\sharp[i_{r-2},i_{r-1}]}_{z_1:=} \cdot \underbrace{\Big((a_r^\sharp \dots a_s^\sharp)[i_{r-1},i_s]\Big)}_{=z} \cdot \underbrace{a_{s+1}^\sharp[i_s,i_{s+1}]\cdots a_k^\sharp[i_{k-1},i_k]}_{z_3:=}$$

  We have $z_1z_2z_3 = \Upsilon_g$, and thus, $z_1 \sqsubseteq \Upsilon_g$ and $z_3 \sqsubseteq \Upsilon_g$. By $\Upsilon_g \sqsubset z$, we have $z_1zz_3 = z$. Thus, $\Upsilon_g \leq z$ which contradicts the assumption $z \in \{\Upsilon_{g+1}, \dots, \Upsilon_{h-1}\}$.

By combining $z \leq \angle_g$ and $z \notin \{\Upsilon_g, \dots, \Upsilon_{h-1}\}$, we obtain $(a_r^\sharp \cdots a_s^\sharp)[i_{r-1}, i_s] = z \sqsubset \Upsilon_g$. Thus, we can apply the inductive hypothesis on $(a_r \dots a_s)[i_{r-1}, i_s]$.

Consequently, there is some $\pi_{r,s} \in i_{r-1} \overset{w_r \dots w_s}{\rightsquigarrow} i_s$ such that we have $\Delta(\theta(\pi_{r,s})) \leq 2^{3g-1}d$ and $\psi(\theta(\pi_{r,s})) \leq (a_r \cdots a_s)^\sharp[i_{r-1}, i_s] = z \sqsubset \Upsilon_g$. By $z \sqsubset \Upsilon_g$, we have $z \sqsubseteq \angle_g$, and by Remark 3.3, we get $z \leq \angle_g$. Thus, $\psi(\theta(\pi_{r,s})) \leq \angle_g$, i.e., $\psi(\theta(\pi_{r,s})) \in \{\Upsilon_0, \dots, \Upsilon_{h-1}, \angle_0, \dots, \angle_g\}$.

We assume such a path $\pi_{r,s}$ for every $1 \leq r \leq s \leq k$ with the above properties.

For every $1 \leq l \leq k$ with $a_l^\sharp[i_{l-1}, i_l] = \Upsilon_g$, let $\pi_l \in i_{l-1} \overset{w_l}{\rightsquigarrow} i_l$ with $\Delta(\theta(\pi_l)) \leq d$ and $\psi(\theta(\pi_l)) = a_l[i_{l-1}, i_l] \leq a_l^\sharp[i_{l-1}, i_l] = \Upsilon_g$. Thus, $\psi(\theta(\pi_l)) \in \{\Upsilon_g, \dots, \Upsilon_{h-1}\}$. Note that there is at least one l with these properties.

Let π be the concatenation of all the paths $\pi_{r,s}$ and π_l "in correct order". Note that by the choice of r and s, there are no two consecutive $\pi_{r,s}$ factors in this concatenation. We have $\pi \in i \overset{w}{\rightsquigarrow} j$. By Lemma 3.17, we have $\Delta(\theta(\pi)) \leq 4 \cdot 2^{3g-1}d = 2^{3g+1}d$.

To prove (b), it remains to show $\psi(\theta(\pi)) \leq \Upsilon_g$. Since $\psi(\theta(\pi_l)) \in \{\Upsilon_g, \dots, \Upsilon_{h-1}\}$ and $\psi(\theta(\pi_{r,s})) \notin \{\angle_{g+1}, \dots, \angle_h\}$ we have $\psi(\theta(\pi)) \in \{\Upsilon_g, \dots, \Upsilon_{h-1}\}$, i.e., $\psi(\theta(\pi)) \leq \Upsilon_g$. ☐

4.6 The Proof of (1)⇒(2) in Theorem 2.1

In this section, we prove (1)⇒(2) in Theorem 2.1.

Proposition 4.9. *Let $I' \subsetneq I \subseteq \langle T \rangle^\sharp$ be ideals of $\langle T \rangle^\sharp$ such that $I \setminus I'$ is a $\mathscr{J}$-class of $\langle T \rangle^\sharp$. For every $k \geq 1$, $(a_1, w_1), \dots, (a_k, w_k) \in \mathcal{V}_{n\times n} \times \Sigma^+$ satisfying*

A1. $a_1, \dots, a_k \in \langle T \rangle^\sharp$,

A2. *for every* $1 \leq l < k$, $a_l a_{l+1} \in I$,

there are $k' \geq 1$, $(a'_1, w'_1), \dots, (a'_{k'}, w'_{k'}) \in \mathcal{V}_{n\times n} \times \Sigma^+$ such that

C1. $a'_1, \dots, a'_{k'} \in \langle T \rangle^\sharp$,

C2. *for every* $1 \leq l < k'$, $a'_l a'_{l+1} \in I'$,

C3. $w'_1 \dots w'_{k'} = w_1 \dots w_k, \quad a_1 \cdots a_k \approx a'_1 \cdots a'_{k'}$ *, and*

C4. $\mathsf{max}_{1 \leq l \leq k'} \Delta'\big(a'_l, w'_l\big) \leq 2^{3h+2} \cdot \mathsf{max}_{1 \leq l \leq k} \Delta'\big(a_l, w_l\big)$

Proof. We denote $d := \mathsf{max}_{1 \leq l \leq k} \Delta'\big(a_l, w_l\big)$.

We factorize $(a_1, w_1), \dots, (a_k, w_k)$. If $a_1 \in I'$, then let $v_1 := (a_1, w_1)$ and $l = 1$. If $a_1 \notin I'$, then let $1 \leq l \leq k$ be the largest integer such that $a_1 \cdots a_l \notin I'$ and set $v_1 = (a_1, w_1), \dots, (a_l, w_l)$. If $l < k$, then we apply the same procedure to $(a_{l+1}, w_{l+1}), \dots, (a_k, w_k)$ and obtain v_2. By repeating this procedure as many times as possible, we achieve sequences $v_1, \dots, v_{k'}$ over $\mathcal{V}_{n\times n} \times \Sigma^+$ with

$v_1 \dots v_{k'} = (a_1, w_1), \dots, (a_k, w_k)$. Note that k' is simply defined as the number of words which we obtain in the factorization.

Let $1 \le l \le k'$ be arbitrary. We define (a'_l, w'_l) from v_l. Let m be the number of pairs in v_l, and denote $v_l = (b_1, u_1), \dots, (b_m, u_m)$. Let $w'_l := u_1 \dots u_m$. Hence, we have $w'_1 \dots w'_{k'} = w_1 \dots w_k$. The definition of a'_l is more involved.

Case 1: $m \le 3$
We set $a'_l = b_1 \dots b_m$. Then, $a'_l \in \langle T \rangle^\sharp$ (C1) and $b_1 \dots b_m \approx a'_l$. By Lemma 4.8(1), we have (C4) for (a'_l, w'_l) since $m \le 3 \le 2^{3h+2}$.

Case 2: $m > 3$, m is even.
We set $a'_l := (b_1 \dots b_m)^\sharp$. For this, we have to ensure that $b_1 \dots b_m \in \mathsf{Reg}(\mathcal{V}_{n\times n})$. However, for (C1), we even have to show $b_1 \dots b_m \in \mathsf{Reg}(\langle T \rangle^\sharp)$. Let $1 \le p < m$ be arbitrary. By (A2), we have $b_p b_{p+1} \in I$ and $b_1 \dots b_m \in I$. Since $m \ge 4$, we have by the definition of v_l and (A2), $b_1 \dots b_m \notin I'$, and in particular $b_p b_{p+1} \notin I'$. Consequently, $b_1 b_2$, $b_3 b_4$, …, $b_{m-1} b_m \in I \setminus I'$ and $b_1 \dots b_m \in I \setminus I'$. Hence, $I \setminus I'$ is a regular $\mathscr{J}$-class of $\langle T \rangle^\sharp$, i.e., $b_1 b_2$, $b_3 b_4$, …, $b_{m-1} b_m$ are a smooth product in $\langle T \rangle^\sharp$. Thus, a'_l is defined and we have $a'_l \in \langle T \rangle^\sharp$ (C1) and $b_1 \dots b_m \approx a'_l$.

We apply Lemma 4.8(2) on $(b_1 b_2, u_1 u_2)$, $(b_3 b_4, u_3 u_4)$, …, $(b_{m-1} b_m, u_{m-1} u_m)$. For every odd $1 \le p < m$, we have $\Delta'(b_p b_{p+1},\ u_p u_{p+1}) \le 2d$. By Lemma 4.8(2), it follows

$$\Delta'(a'_l, u_1 \dots u_m) \ \le\ \Delta'\big((b_1 \dots b_m)^\sharp,\ u_1 \dots u_m\big) \ \le\ 2^{3h} d,$$

i.e., $(a'_l, u_1 \dots u_m)$ satisfies (C4).

Case 3: $m > 3$, m is odd.
We proceed as in case 2, but we consider the sequence $(b_1 b_2 b_3, u_1 u_2 u_3)$, $(b_4 b_5, u_4 u_5)$, …, $(b_{m-1} b_m, u_{m-1} u_m)$. We get $\Delta'(a'_l, u_1 \dots u_m) \le 3 \cdot 2^{3h-1} d$, i.e., $(a'_l, u_1 \dots u_m)$ satisfies (C4).

Since $b_1 \cdots b_m \approx a'_l$ in each case, we have $a_1 \cdots a_k \approx a'_1 \cdots a'_{k'}$ which completes (C3).

It remains to show (C2). Let $1 \le l < k'$. As above, we denote $v_l = (b_1, u_1), \dots, (b_m, u_m)$. Moreover, we denote $v_{l+1} = (\hat{b}_1, \hat{u}_1), \dots, (\hat{b}_{\hat{m}}, \hat{u}_{\hat{m}})$. We have $a'_l = b_1 \dots b_m$ or $a'_l = (b_1 \dots b_m)^\sharp$, and hence, $a'_l \le_{\mathscr{L}} b_1 \dots b_m$, and similarly, $a'_{l+1} \le_{\mathscr{R}} \hat{b}_1 \dots \hat{b}_{\hat{m}}$ (cf. Remark 3.2). Consequently, $a'_l a'_{l+1} \le_{\mathscr{J}} b_1 \dots b_m \hat{b}_1 \dots \hat{b}_{\hat{m}}$. By our factorization method to obtain v_l and v_{l+1}, above, we have either $b_1 \in I'$, $\hat{b}_1 \in I'$, or $b_1 \dots b_m \hat{b}_1 \in I'$. To sum up, $a'_l a'_{l+1} \in I'$ (C2). □

Proof of (1)⇒(2) in Theorem 2.1. We assume that for every $a \in \langle T \rangle^\sharp$, we have $I \cdot a \cdot F \ne \omega$, and we show that $\mathcal{A}$ is limited. Let $w \in L(\mathcal{A})$ be an arbitrary, non-empty word. Let $k := |w|$ and denote $w = c_1 \dots c_k$.

Let y be the number of $\mathscr{J}$-classes of $\langle T \rangle^\sharp$. Let $\langle T \rangle^\sharp = I_1 \supsetneq I_2 \supsetneq \cdots \supsetneq I_y \supsetneq I_{y+1} = \emptyset$ be ideals of $\langle T \rangle^\sharp$ such that for every $1 \le l \le y$ the set $I_l \setminus I_{l+1}$ is a $\mathscr{J}$-class of $\langle T \rangle^\sharp$.

Consider the sequence $\big(\Psi(c_1), c_1\big)$, …, $\big(\Psi(c_k), c_k\big)$. We apply Proposition 4.9 inductively y times for $I_0, \dots, I_y$ on this sequence. Initially, $I = \langle T \rangle^\sharp$, and hence, (A2) is satisfied. Clearly, $\Psi(c_1), \dots \Psi(c_k) \in T \subseteq \langle T \rangle^\sharp$, i.e., (A1) is satisfied. In each application of Proposition 4.9, (C1) and (C2) provide (A1) and (A2) for the next application. In the last application, $I' = \emptyset$, and thus, (C2) implies $k' = 1$. Hence, we obtain a single pair (a, w), and we have by (C3, C4), $a \approx \Psi(c_1) \dots \Psi(c_m) = \Psi(w)$ and $a \in \langle T \rangle^\sharp$. For every $1 \le l \le k$, we have $\Delta'\big(\Psi(c_l), c_l\big) \le 1$. By (C4) and since $y \le |\mathcal{V}_{n\times n}| = (2h+3)^{n^2}$, we get

$$\Delta'\big(a, w\big) \ \le\ 2^{(3h+2)(2h+3)^{n^2}}$$

Since $w \in L(\mathcal{A})$, we have $I \cdot \Psi(w) \cdot F \neq \infty$, and since $a \approx \Psi(w)$, we get $I \cdot a \cdot F \neq \infty$. Since $a \in \langle T \rangle^\sharp$, we get $I \cdot a \cdot F \neq \omega$. Consequently, $I \cdot a \cdot F < \omega$. Hence, there is a successful path π in $\mathcal{A}$ with the label w and $\Delta(\pi) \leq 2^{(3h+2)(2h+3)^{n^2}}$, i.e., $\mathcal{A}$ is limited. □

4.7 Bibliographic Remarks

Section 4 was developed by the author inspired by techniques from I. Simon and H. Leung [29, 30, 50, 53] but in particular [34].

The notion of stabilization is a joint generalization of various stabilizations used by I. Simon, H. Leung, and the author [29, 30, 34, 51, 53, 22, 25].

The techniques in Section 4.2 originate from the author, but similar ideas were already known for distance automata [53, 34].

Section 4.3 originates from the author but similar techniques are due to H. Leung for distance automata [29, 30, 34]. The proofs of Lemma 4.4 and Proposition 4.5 originate from the author.

Lemmas 4.6 and 4.7 were shown by a the author as a generalization of simpler lemmas by H. Leung for matrices over $\mathcal{R}$. Proposition 4.8 and its proof originate from the author.

In [34], H. Leung gave an improved variant of his approach to the limitedness problem of distance automata. Unfortunately, the most interesting part ($\Psi(< T >^c) \subseteq \Psi(T)^{+,\#}$ in Theorem 1 in [34, p. 98]) is just roughly sketched. The author completed the missing parts in [34] for distance automata and adapted and generalized the construction to nested distance desert automata. In this adaptation and generalization, the author developed Proposition 4.9, its proof, and the proof of (1)$\Rightarrow$(2) in Theorem 2.1. Consequently, Section 4.6 originates mainly from the author although his development of Section 4.6 started as a completion of rough sketches from [34].

5 On the Complexity

5.1 The Stabilization Hierarchy

Let $T \subseteq \mathcal{V}_{n\times n}$ and set $T_0 := \langle T \rangle$. For every $p \in \mathbb{N}$, let

$$T_{p+1} := \Big\langle\, T_p \,\cup\, \big\{ e^\sharp \,\big|\, e \,\in\, \mathsf{E}(T_p) \big\} \,\Big\rangle$$

We call $T_0 \subseteq T_1 \subseteq T_2 \ldots$ the *stabilization hierarchy of* T. Moreover, it is easy to see that $\langle T \rangle^\sharp = \bigcup_{p\geq 0} T_p$. Because for every $p \geq 0$, T_p is a subset of the finite set $\mathcal{V}_{n\times n}$, we have $T_{|\mathcal{V}_{n\times n}|} = T_{|\mathcal{V}_{n\times n}|+1}$, and hence, $\langle T \rangle^\sharp = T_{|\mathcal{V}_{n\times n}|}$.

A key question for the complexity of limitedness of nested distance desert automata is: at which level does the stabilization hierarchy collapse? This question was already raised by H. Leung in the framework of distance automata in 1987 [29].

Recall that $\mathcal{R} = \{\curlyvee_0, \angle_1, \omega, \infty\}$. For $T \subseteq \mathcal{R}_{n\times n}$, H. Leung conjectured $\langle T \rangle^\sharp = T_{n^2}$ [29, p. 38]. In [51, p. 112] it was conjectured that there is a polynomial $B : \mathbb{N} \to \mathbb{N}$ such that $\langle T \rangle^\sharp = T_{B(n)}$ for every $T \subseteq \mathcal{R}_{n\times n}$. In [32, p. 522], the existence of such a polynomial B was again considered as an open question. This question was very important, because the existence of such a polynomial B implies that limitedness of distance automata is decidable in PSPACE [29, 32].

However, in 1998, H. Leung suggested another strategy. He mentioned that limitedness of distance automata is decidable in PSPACE if there is some polynomial $C : \mathbb{N} \to \mathbb{N}$ such that every n-state distance automaton is either limited by $2^{C(n)}$ or unlimited [34]. Indeed, K. Hashiguchi showed that this assertion is true for $C(n) = 4n^3 + n\,\mathrm{ld}(n+2) + n \;\leq\; 4n^3 + n^2 + 2n$ [16, 17, 18]. H. Leung and V. Podolskiy improved this bound to $C(n) = 3n^3 + n\,\mathrm{ld}\,n + n - 1$ [35], and hence, limitedness of distance automata is decidable in PSPACE.

However, it remained open whether there is a polynomial B for the collapse of the stabilization hierarchy. Let us mention that H. Leung showed for every $n \geq 2$ some set $T \subseteq \mathcal{R}_{n\times n}$ such that $T_{n-2} \subsetneq T_{n-1} = \langle T \rangle^\sharp$, i.e., setting $B(n) := n-2$ is not sufficient [32].

Below, we will positively answer H. Leung's conjecture by showing $T_n = \langle T \rangle^\sharp$ in Corollary 5.6(2).

5.2 Index Classes

Let $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ and $0 \leq g \leq h$. We define a relation $\sim_{e,g}$ on $\{1, \ldots, n\}$ by setting

$$i \sim_{e,g} j \qquad :\Longleftrightarrow \qquad e[i,j] \leq \angle_g \;\text{ and }\; e[j,i] \leq \angle_g$$

for every i, j. Clearly, $\sim_{e,g}$ is symmetric, and since e is idempotent, $\sim_{e,g}$ is transitive. If for some i, there is a j such that $i \sim_{e,g} j$, then we have $i \sim_{e,g} i$. Consequently, the restriction of $\sim_{e,g}$ to the set

$$Z_{e,g} := \big\{ i \,\big|\, \text{there is some } j \text{ such that } i \sim_{e,g} j \big\}$$

is reflexive, i.e., $\sim_{e,g}$ is an equivalence relation on $Z_{e,g}$. By *equivalence class of* $\sim_{e,g}$ we mean an equivalence class of $\sim_{e,g}$ on $Z_{e,g}$. For every $i \in Z_{e,g}$, we denote by $[i]_{e,g}$ the equivalence class of i. We denote by $\mathsf{Cl}(e,g)$ the set of equivalence classes of $\sim_{e,g}$.

Lemma 5.1. *Let $e, f \in \mathsf{E}(\mathcal{V}_{n\times n})$ such that $e \geq_{\mathscr{J}} f$ and $0 \leq g \leq h$. We have $\big|\mathsf{Cl}(e,g)\big| \;\geq\; \big|\mathsf{Cl}(f,g)\big|$.*

Proof. Let $a, b \in \mathcal{V}_{n\times n}$ such that $aeb = f$. We assume $ae = a$ and $eb = b$. If a and b do not satisfy these conditions, then we proceed the proof for $a' = ae$ and $b' = eb$.

We construct a partial surjective mapping $\beta : \mathsf{Cl}(e, g) \dashrightarrow \mathsf{Cl}(f, g)$. The mapping β depends on the choice of a and b. For every i, j with $i \sim_{e,g} i$ and $j \sim_{f,g} j$ satisfying $a[j, i] \cdot e[i, i] \cdot b[i, j] \le \angle_g$, we set $\beta([i]_{e,g}) := [j]_{f,g}$. To complete the proof, we have to show that β is well defined and that β is indeed surjective.

We show that β is well defined. Let i, i' such that $i \sim_{e,g} i$ and $i' \sim_{e,g} i'$. Moreover, let j, j' such that $j \sim_{f,g} j$ and $j' \sim_{f,g} j'$. Assume $a[j, i] \cdot e[i, i] \cdot b[i, j] \le \angle_g$ and $a[j', i'] \cdot e[i', i'] \cdot b[i', j'] \le \angle_g$. Thus, $\beta([i]_{e,g}) = [j]_{f,g}$ and $\beta([i']_{e,g}) = [j']_{f,g}$. To show that β is well defined, we have to show that if $[i]_{e,g} = [i']_{e,g}$, then we have $[j]_{f,g} = [j']_{f,g}$. Assume $[i]_{e,g} = [i']_{e,g}$, i.e., $i \sim_{e,g} i'$. Hence, $e[i, i'] \le \angle_g$. Above, we assumed $a[j, i] \cdot e[i, i] \cdot b[i, j] \le \angle_g$, and thus, $a[j, i] \le \angle_g$. Similarly, $b[i', j'] \le \angle_g$. Consequently, $a[j, i] \cdot e[i, i'] \cdot b[i', j'] \le \angle_g$, i.e., $f[j, j'] = (aeb)[j, j'] \le \angle_g$. By symmetry, we achieve $f[j', j] \le \angle_g$, and hence, $j \sim_{f,g} j'$.

We show that β is surjective. Let j such that $j \sim_{f,g} j$. We have to show some i such that $\beta([i]_{e,g}) = [j]_{f,g}$. Since $j \sim_{f,g} j$, we have $f[j, j] \le \angle_g$. Since $f = aeb$, there are k, l such that $a[j, k] \cdot e[k, l] \cdot b[l, j] \le \angle_g$, and in particular, $e[k, l] \le \angle_g$. By Lemma 3.11, there is some i such that $e[k, i] \cdot e[i, i] \cdot e[i, l] = e[k, l] \le \angle_g$, and in particular, $e[i, i] \le \angle_g$. We have $a[j, i] = (ae)[j, i] \le a[j, k] \cdot e[k, i] \le \angle_g$, and $b[i, j] = (eb)[i, j] = e[i, l] \cdot b[l, j] \le \angle_g$. To sum up, $a[j, i] \cdot e[i, i] \cdot b[i, j] \le \angle_g$, and hence, $\beta([i]_{e,g}) = [j]_{f,g}$. □

Let $e \in \mathsf{E}(\mathcal{V}_{n\times n})$, $0 \le g \le h$ and i, j be arbitrary. By Remark 4.1(3), we easily observe that $i \sim_{e^\sharp,g} j$ implies $i \sim_{e,g} j$. In particular, we have $Z_{e,g} \supseteq Z_{e^\sharp,g}$.

Lemma 5.2. *Let $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ and $0 \le g \le h$. We have $\mathsf{Cl}(e, g) \supseteq \mathsf{Cl}(e^\sharp, g)$.*

Proof. Let i be such that $i \sim_{e^\sharp,g} i$. We show $[i]_{e^\sharp,g} = [i]_{e,g}$.

For every j with $i \sim_{e^\sharp,g} j$, we have by Remark 4.1(3), $i \sim_{e,g} j$. Hence, $[i]_{e^\sharp,g} \subseteq [i]_{e,g}$.

Conversely, let $j \in [i]_{e,g}$. Hence, $e[i, j] \le \angle_g$. Since $i \sim_{e^\sharp,g} i$, we have $e^\sharp[i, i] \le \angle_g$. To sum up,

$$e^\sharp[i, j] \;=\; (e^\sharp e)[i, j] \;\le\; e^\sharp[i, i] \cdot e[i, j] \;\le\; \angle_g \cdot \angle_g \;=\; \angle_g,$$

and by symmetry, $e^\sharp[j, i] \le \angle_g$, i.e., $i \sim_{e^\sharp,g} j$. Hence, $j \in [i]_{e^\sharp,g}$. □

Lemma 5.3. *Let $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ and assume $e \ne e^\sharp$. There is some $0 \le g \le h$ such that we have $\mathsf{Cl}(e, g) \supsetneq \mathsf{Cl}(e^\sharp, g)$ and for some l, $e[l, l] = \angle_g$.*

Proof. Let i, j such that $e[i, j] \ne e^\sharp[i, j]$. By Lemma 3.11, there is some l such that $e[i, j] = e[i, l] \cdot e[l, l] \cdot e[l, j]$. By contradiction, assume $e[l, l] = e^\sharp[l, l]$. Hence,

$$e^\sharp[i, j] \;=\; (ee^\sharp e)[i, j] \;\le\; e[i, l] \cdot e^\sharp[l, l] \cdot e[l, j] \;=\; e[i, l] \cdot e[l, l] \cdot e[l, j] \;=\; e[i, j],$$

i.e., $e^\sharp[i, j] = e[i, j]$ which is a contradiction. Consequently, $e[l, l] < e^\sharp[l, l]$.

By Remark 4.1(4, 5, 6), we have $e[l, l] \notin \{\curlyvee_0, \ldots, \curlyvee_{h-1}, \omega, \infty\}$, and thus, $e[l, l] \in \{\angle_0, \ldots, \angle_h\}$. Let $0 \le g \le h$ such that $e[l, l] = \angle_g$. We have $e^\sharp[l, l] > \angle_g$. Consequently, $l \sim_{e,g} l$, but we do not have $l \sim_{e^\sharp,g} l$. Thus, $l \in Z_{e,g}$ but $l \notin Z_{e^\sharp,g}$. Hence, there is a class $[l]_{e,g}$ in $\mathsf{Cl}(e, g)$, but there is no class $[l]_{e^\sharp,g}$ in $\mathsf{Cl}(e^\sharp, g)$. In combination with Lemma 5.2, we obtain $\mathsf{Cl}(e, g) \supsetneq \mathsf{Cl}(e^\sharp, g)$. □

5.3 The Collapse of the Stabilization Hierarchy

We fix some $T \subseteq \mathcal{V}_{n\times n}$ for Section 5.3. We define

$$\angle(T) := \{\angle_g \mid a[i,j] = \angle_g \text{ for some } a \in T \text{ and } i,j\}.$$

Since $T \subseteq \langle T\rangle^\sharp$, we have $\angle(T) \subseteq \angle(\langle T\rangle^\sharp)$, $e \in \mathsf{E}(\mathcal{V}_{n\times n})$. Let $a, b \in \mathcal{V}_{n\times n}$. If there is some $\angle_g$ in ab, then there is some $\angle_g$ in a or in b. If there is some $\angle_g$ in $e^\sharp$, then there is some $\angle_g$ in e. Hence, $\angle(T) = \angle(\langle T\rangle^\sharp)$.

For every $e \in \mathsf{E}(\mathcal{V}_{n\times n})$, we set

$$\mathsf{cls}(e) := \sum_{\angle_g \in \angle(T)} |\mathsf{Cl}(e,g)|.$$

Note that $\mathsf{cls}(e)$ depends on the underlying set T. For example, let e be the matrix in which every entry is Υ_0. For every $0 \le g \le h$, the set $\{1,\ldots,n\}$ is the only equivalence class of $\sim_{e,g}$. Then, we have $|\mathsf{Cl}(e,g)| = 1$ and $\mathsf{cls}(e) = |\angle(T)|$.

For every $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ and $0 \le g \le h$, we have $|\mathsf{Cl}(e,g)| \le n$, and hence, $\mathsf{cls}(e) \le |\angle(T)|n$. By Lemma 5.2 and 5.3, we have $\mathsf{cls}(e) \le \mathsf{cls}(e^\sharp)$ for every $e \in \langle T\rangle^\sharp \cap \mathsf{E}(\mathcal{V}_{n\times n})$. If $e \ne e^\sharp$, then we even have $\mathsf{cls}(e) < \mathsf{cls}(e^\sharp)$. This observation allows us to show that the stabilization hierarchy of T collapses at level $|\angle(T)|n$.

Lemma 5.4. *Let $T \subseteq \mathcal{V}_{n\times n}$ and $p \ge 1$. For every $e \in T_p \setminus T_{p-1}$ with $e \in \mathsf{E}(\mathcal{V}_{n\times n})$, we have*

$$\mathsf{cls}(e) \le |\angle(T)|n - p.$$

In the particular case $p > |\angle(T)|n$, Lemma 5.4 implies that there is no idempotent in $T_p \setminus T_{p-1}$.

Proof. For a more lucid presentation of the proof, we set $T_{-1} := \emptyset$ and show the lemma for $p \ge 0$. We proceed by induction on p. For $p = 0$, the assertion is obvious.

Let $p \ge 0$. We show the claim for $p+1$. Let $e \in T_{p+1} \setminus T_p$ with $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ be arbitrary. By the definition of T_{p+1}, there are some $k \ge 1$ and $a_1,\ldots,a_k \in \mathcal{V}_{n\times n}$ such that $e = a_1 \ldots a_k$ and for every $1 \le i \le k$, we have $a_i \in T_p$ or $a_i = e_i^\sharp$ for some $e_i \in \mathsf{E}(T_p)$. Since $e \in T_{p+1} \setminus T_p$, there is at least one $1 \le i \le k$ such that $a_i = e_i^\sharp$ for some $e_i \in \mathsf{E}(T_p)$ such that $e_i^\sharp \notin T_p$. By $e_i^\sharp \notin T_p$, we have $e_i \notin T_{p-1}$. Hence, $e_i \in T_p \setminus T_{p-1}$.

By induction, we have $\mathsf{cls}(e_i) \le |\angle(T)|n - p$. Since $e_i \ne e_i^\sharp$, we obtain by Lemma 5.2 and 5.3 $\mathsf{cls}(e_i^\sharp) < \mathsf{cls}(e_i)$. Since $e \le_{\mathscr{J}} e_i^\sharp$, we obtain by Lemma 5.1, $\mathsf{cls}(e) \le \mathsf{cls}(e_i^\sharp)$. To sum up, we have $\mathsf{cls}(e) \le |\angle(T)|n - (p+1)$. ◻

Proposition 5.5. *Let $T \subseteq \mathcal{V}_{n\times n}$. We have $T_{|\angle(T)|n} = T_{|\angle(T)|n+1}$, i.e., the stabilization hierarchy of T collapses at level $|\angle(T)|n$, and in particular, $T_{|\angle(T)|n} = \langle T\rangle^\sharp$.*

Proof. Let $p := |\angle(T)|n$. By contradiction, let $e \in \mathsf{E}(T_p)$ such that $e^\sharp \notin T_p$. By the definition of T_{p+1}, we have $e^\sharp \in \mathsf{E}(T_{p+1})$. By Lemma 5.4, we have $\mathsf{cls}(e) \le |\angle(T)|n - (p+1) = -1$, which is a contradiction. ◻

For lucidity, we state the following corollary:

Corollary 5.6.

1. *Let $h \ge 1$ and $T \subseteq \mathcal{V}_{n\times n}$. We have $T_{(h+1)n} = \langle T\rangle^\sharp$.*

2. *For every subset $T \subseteq \mathcal{R}_{n\times n}$, we have $T_n = \langle T \rangle^\sharp$.*

Proof. Assertion (1) follows from Proposition 5.4 because $T_0 \subseteq T_1 \subseteq T_2 \subseteq \ldots$ and $|\angle(T)| \le h+1$. Since $\mathcal{R} = \{\Upsilon_0, \angle_1, \omega, \infty\}$, we have $|\angle(T)| \le 1$. Hence, (2) follows from Proposition 5.4 □

As already mentioned, H. Leung conjectured in 1987 [29] $T_{n^2} = \langle T \rangle^\sharp$ for every $T \subseteq \mathcal{R}_{n\times n}$. Corollary 5.6(2) is a positive answer to this conjecture, because $T_n \subseteq T_{n^2} \subseteq \langle T \rangle^\sharp$.

We complete the proof of Theorem 2.1 by showing (2)⇒(3).

Proof of (2)⇒(3) in Theorem 2.1. For every $a \in T_0$, we can construct a $\sharp$-expression r which does not contain any $\sharp$ such that $\tau(r) = a$. By induction, we can construct for every $p \ge 0$ and every $a \in T_p$ a typed $\sharp$-expression r such that $\tau(r) = a$ and the $\sharp$-height of r is at most p.

Let $a \in \langle T \rangle^\sharp$ such that $I \cdot a \cdot F = \omega$. By Corollary 5.6(1), $a \in T_{(h+1)|Q|}$, i.e., there is a $\sharp$-expression r such that $\tau(r) = a$ and the $\sharp$-height of r is at most $(h+1)|Q|$. By Proposition 4.3, r proves (3). □

5.4 A Nondeterministic PSPACE Algorithm

Again let $T \subseteq \mathcal{V}_{n\times n}$. We define

$$\mathsf{ent}(T) := \{ a[i,j] \mid \text{ for some } a \in T \text{ and } i,j \}.$$

We have $\mathsf{ent}(T) \subseteq \mathsf{ent}(\langle T \rangle^\sharp) \subseteq \mathsf{ent}(T) \cup \{\omega\}$.

Lemma 5.7. *Let $T \subseteq \mathcal{V}_{n\times n}$.*

1. *For every $a \in T_0$, there are $1 \le m \le |\mathsf{ent}(T)|^{n^2}$ and $a_1, \ldots, a_m \in \mathcal{V}_{n\times n}$ such that $a = a_1 \ldots a_m$ and for every $1 \le k \le m$, $a_k \in T$.*
2. *Let $p \ge 1$. For every $a \in T_p$, there are $1 \le m \le (|\mathsf{ent}(T)|+1)^{n^2}$ and $a_1, \ldots, a_m \in \mathcal{V}_{n\times n}$ such that $a = a_1 \ldots a_m$ and for every $1 \le k \le m$,*
   - *(a) $a_k \in T_{p-1}$ or*
   - *(b) there is some $e_k \in \mathsf{E}(T_{p-1})$ such that $a_k = e_k^\sharp$.*

Proof. (1) Let $a \in T_0$. Since $T_0 = \langle T \rangle$, there are $m \ge 1$ and $a_1, \ldots, a_m \in T$ such that $a = a_1 \ldots a_m$. By a counting and cancellation argument, we can assume $m \le |\mathsf{ent}(T)|^{n^2}$.

(2) We apply the definition of T_p. We can assume $m \le (|\mathsf{ent}(T)|+1)^{n^2}$, because we have $a \in \langle T \rangle^\sharp \subseteq (\mathsf{ent}(T) \cup \{\omega\})_{n\times n}$. □

We give a nondeterministic algorithm to decide limitedness of nested distance desert automata. The key of the algorithm is the function `guess`, below. The input of `guess` are a non-empty set $T \subseteq \mathcal{V}_{n\times n}$ and an integer $p \ge 0$. Below, we will show that `guess` returns some matrix in $a \in \langle T \rangle^\sharp$.

We assume some function `guessbool` which returns nondeterministically `true` or `false`. We also assume some function `guessnum` whose argument is a integer. For every $k \ge 1$, `guessnum(`k`)` returns nondeterministically an integer between 1 and k. Moreover, we assume a function `choose`. The input of `choose` is a non-empty set of matrices in $\mathcal{V}_{n\times n}$ and it returns nondeterministically some matrix from the input set.

```
function guess(T, p)
a:= 1_{V_{n×n}}
for k:=1 to guessnum((|ent(T)| + 1)^{n^2}) do begin
```

```
    if p = 0 then a:= a·choose(T)
    else begin
      b:= guess(T, p − 1)
      if b·b = b and guessbool then b:= b♯
      a:= a·b
    end
  end
  return a
```

Proposition 5.8. *Let $T \subseteq \mathcal{V}_{n\times n}$ be non-empty, $p \geq 0$, and $a \in \mathcal{V}_{n\times n}$. There is a run of* `guess(`$T$`,`$p$`)` *which returns the matrix a iff $a \in T_p$.*

Proof. Let $p = 0$. Clearly, `guess(`T`,`0`)` returns a product of matrices in T. Conversely, by Lemma 5.7(1), there is for every $a \in T_0$ some run of `guess(`T`,`0`)` which returns a.

Let $p \geq 1$ and assume that the assertion is true for $p-1$. We prove the assertion for p.

$\ldots \Rightarrow \ldots$ Every run of `guess(`T`,`p`)` returns a product of matrices $a_1, \ldots, a_m \in \mathcal{V}_{n\times n}$ for some $1 \leq m \leq |\mathsf{ent}(T)+1|^{n^2}$ whereas for every $1 \leq k \leq m$, a_k is a result of `guess(`T`,`$p-1$`)` or $a_k = e_k^\sharp$ for some result e_k of `guess(`T`,`$p-1$`)` with $e_k \in \mathsf{E}(\mathcal{V}_{n\times n})$. By induction, every result of `guess(`T`,`$p-1$`)` belongs to T_{p-1}. By the definition of T_p, the result of `guess(`T`,`p`)` belongs to T_p.

$\ldots \Leftarrow \ldots$ Let $a \in T_p$. Let $1 \leq m \leq |\mathsf{ent}(T)+1|^{n^2}$ and $a_1, \ldots, a_m \in \mathcal{V}_{n\times n}$ as in Lemma 5.7(2). We show a run of `guess(`T`,`p`)` which returns a. We assume that `guessnum(`$|\mathsf{ent}(T)+1|^{n^2}$`)` returns m. Let $1 \leq k \leq m$ be arbitrary. We consider the k-th run of the loop. If $a_k \in T_{p-1}$, then we assume by the inductive hypothesis that `guess(`T`,`$p-1$`)` returns a_k and `guessbool` returns false. If $a_k \notin T_{p-1}$, then $a_k = e_k^\sharp$ for some $e_k \in \mathsf{E}(T_{p-1})$, and we assume by the inductive hypothesis that `guess(`T`,`$p-1$`)` returns b_k and `guessbool` returns true. By an induction on k, one can show that the value of a after the k-th run of the loop is $a_1 \cdots a_k$, and thus, `guess(`T`,`p`)` returns a. □

We show that limitedness of nested distance desert automata is decidable in PSPACE. We will show PSPACE-hardness in Section 5.5.

Proof of Theorem 2.2. We sketch a nondeterministic algorithm which decides limitedness of nested distance desert automata. Let $\mathcal{A} = [Q, E, I, F, \theta]$ be a nested distance desert automaton. At first, the algorithm constructs the set $\Psi(\Sigma) \in \mathcal{V}_{n\times n}$. Let us denote $n := |Q|$, $T := \Psi(\Sigma)$ and $p := |\angle(T)|n$. The algorithm computes $I \cdot$ `guess(`T`,`p`)` $\cdot F$. If the result is ω, then the algorithm returns "$\mathcal{A}$ is not limited", otherwise, the computation fails.

If $\mathcal{A}$ is not limited, then there is a matrix $a \in \langle T\rangle^\sharp$ such that $I \cdot a \cdot F = \omega$. By Proposition 5.5, $a \in T_p$. By Proposition 5.8, there is a run of `guess(`T`,`p`)` which returns a. Hence, there is a run on which the algorithm returns "$\mathcal{A}$ is not limited".

If there is a run on which the algorithm returns "$\mathcal{A}$ is not limited", then there is a run of `guess(`T`,`p`)` which returns some matrix $a \in \mathcal{V}_{n\times n}$ for which $I \cdot a \cdot F = \omega$. By Proposition 5.8 and $T_p \subseteq \langle T\rangle^\sharp$, we have $a \in \langle T\rangle^\sharp$, i.e., $\mathcal{A}$ is unlimited.

The algorithm requires $n^2\mathrm{ld}(|\mathsf{ent}(T)|+1)$ bits to store the value $(\mathsf{ent}(T)+1)^{n^2}$ for `guessnum` calls. It is not necessary to store the set T explicitly. An implementation of `choose(`T`)` can non-nondeterministically choose some letter from Σ and retrieve the corresponding matrix from the automaton.

For a run of `guess(`T`,`p`)`, one has to store the counter k, the matrices a and b, and a temporary matrix to compute matrix multiplication, stabilization, and the comparison $b \cdot b = b$. Moreover, the recursive call of $\texttt{guess}(T, p-1)$ requires space. One can store k in $\mathrm{ld}(|\mathsf{ent}(T)+1|)^{n^2} = n^2\mathrm{ld}(|\mathsf{ent}(T)+1|)$ bits, and one can store each matrix in $n^2\mathrm{ld}(|\mathsf{ent}(T)+1|)$ bits. Hence, a run of $\texttt{guess}(T, p)$ requires $4n^2\mathrm{ld}(|\mathsf{ent}(T)+1|)$ bits and additionally space for the recursive call of $\texttt{guess}(T, p-1)$. By an induction on p, one can show that a run of $\texttt{guess}(T, p)$ requires $(p+1)4n^2\mathrm{ld}(|\mathsf{ent}(T)+1|) = \big(|\angle(T)|n+1\big)4n^2\mathrm{ld}\big(|\mathsf{ent}(T)+1|\big) \in \mathcal{O}\big(|\angle(T)|n^3\mathrm{ld}\,\mathsf{ent}(T)\big)$ space.

We have $|\mathsf{ent}(T)| \leq 2h+2$ and $|\angle(T)| \leq h+1$. For fixed h, our nondeterministic algorithm requires $\mathcal{O}(n^3)$ space. If h is not fixed, we still have $|\mathsf{ent}(T)| \leq |E|$ and $|\angle(T)| \leq |E|$. We can assume $n \leq 2|E|$. Thus, our nondeterministic algorithm requires $\mathcal{O}\big(|E|^3\mathrm{ld}|E|\big)$ space.

By Savitch's theorem, limitedness of nested distance desert automata is decidable in deterministic polynomial space. For fixed h, the space is polynomial in the number of states. For arbitrary h, the space is polynomial in the number of transitions. □

5.5 PSPACE-Hardness of the Limitedness Problem

We show that limitedness is PSPACE-hard even for very restricted nested distance desert automata.

Proposition 5.9. *Let $g, h \in \mathbb{N}$ be arbitrary. Limitedness of nested distance desert automata in which each transition is marked by $\curlyvee_g$ or $\angle_h$ is PSPACE-hard.*

Proof. We follow the same idea as H. Leung's proof for PSPACE-hardness of limitedness of distance automata [29, 30]. Let $\mathcal{A} = [Q, E, I, F]$ be a nondeterministic automaton. The problem whether $L(\mathcal{A}) = \Sigma^*$ is known to be PSPACE-hard [20]. We construct a nested distance desert automaton which is limited iff $L(\mathcal{A}) = \Sigma^*$.

Let $c \notin \Sigma$ be a new letter. We can construct an automaton $\mathcal{A}'$ which accepts $L(\mathcal{A})c^+$ by adding just one state to $\mathcal{A}$. We mark every transition in $\mathcal{A}'$ by $\curlyvee_g$. For every $w \in L(\mathcal{A})$, $k \geq 1$, we have $\Delta_{\mathcal{A}'}(wc^k) = 0$.

By adding two more states to $\mathcal{A}'$, we can construct a nested distance desert automaton $\mathcal{A}''$ which accepts $\Sigma^* c^+$. We mark the new transitions by $\angle_h$. For every $w \in \Sigma^*$, $k \geq 1$, we have $\Delta_{\mathcal{A}''}(wc^k) = k$ if $w \notin L(\mathcal{A})$ but $\Delta_{\mathcal{A}''}(wc^k) = 0$ if $w \in L(\mathcal{A})$. Obviously, $\mathcal{A}''$ is limited iff $L(\mathcal{A}) = \Sigma^*$ and the size of $\mathcal{A}''$ is polynomial in the size of $\mathcal{A}$. □

5.6 Bibliographic Remarks

The stabilization hierarchy was already mentioned by H. Leung for matrices over $\mathcal{R}$, i.e., in the framework of distance automata [29, 32]. As explained in Section 5.1, H. Leung raised the question at which level the stabilization hierarchy collapses in his PhD thesis in 1987 [29] and later in [32].

The key ideas of Section 5.2 and 5.3 are not generalized from known results for distance automata. These techniques are entirely new even for distance automata. In particular, the idea to introduce the relation $\sim_{e,g}$ to answer H. Leung's question originates from the author as well as Lemmas 5.1, 5.2, 5.3, 5.4, Proposition 5.5, Corollary 5.6 and their proofs.

Section 5.4 originates from the author, but the existence of such an algorithm for distance automata was already conjectured by H. Leung in [29, 32].

The proof of Proposition 5.9 in Section 5.5 is a straightforward adaptation of H. Leung's proof for the same result for distance automata [29, 30].

6 On the Star Height Problem

The aim of this section is to prove Theorem 2.3 and 2.4.

During Section 6, let $h \in \mathbb{N}$ be some integer. Moreover, let M be some finite monoid and $\eta : \Sigma^* \to M$ be some surjective homomorphism.

In order to prove Theorem 2.3 we develop techniques to determine whether the star height of languages $\eta^{-1}(P)$ for $P \subseteq M$ is less than h. At first, we explain our strategy.

We follow R.S. Cohen [4] and define a particular class of rational expressions, the *string expressions*. There are two interesting parameters concerning the size of a string expression: the star height and the degree. We borrow from [4] that for every recognizable language L, there is a string expression r such that $\mathsf{sh}(L) = \mathsf{sh}(r)$.

Now, let $P \subseteq M$. We are interested in whether the star height of $\eta^{-1}(P)$ is less than or equal to h. For this, we construct a sequence of string expressions $r_1, r_2, \ldots$ The expressions $r_1, r_2, \ldots$ are of a star height of at most h. For every $d \geq 1$, the degree of r_d is at most d.

We construct these expressions such that we have $L(r_1) \subseteq L(r_2) \subseteq L(r_3) \subseteq \cdots \subseteq \eta^{-1}(P)$ and $\bigcup_{d \geq 1} L(r_d) = \eta^{-1}(P)$. Moreover, we construct $r_1, r_2, \ldots$ such that for every $d \geq 1$, r_d is the most general expression of star height h and degree d inside $\eta^{-1}(P)$. In other words, if r is any string expression of a star height at most h and a degree of at most d and $L(r) \subseteq \eta^{-1}(P)$, then we have $L(r) \subseteq L(r_d)$.

If the star height of $\eta^{-1}(P)$ is less than or equal to h, then $\eta^{-1}(P)$ is generated by some string expression r such that $\mathsf{sh}(r) \leq h$. Let d be the degree of r. As seen above, we have

$$\eta^{-1}(P) \;=\; L(r) \;\subseteq\; L(r_d) \;\subseteq\; L(r_{d+1}) \;\subseteq\; L(r_{d+2}) \;\subseteq\; \cdots \;\subseteq\; \eta^{-1}(P),$$

i.e., $L(r_{d'}) = \eta^{-1}(P)$ for every $d' \geq d$. Thus, the hierarchy $L(r_1) \subseteq L(r_2) \subseteq \cdots$ collapses at level d.

Conversely, if the hierarchy $L(r_1) \subseteq L(r_2) \subseteq \cdots$ collapses at some level d, then $L(r_d) = \eta^{-1}(P)$, and hence, $\eta^{-1}(P)$ is of a star height of at most h.

Hence, the hierarchy $L(r_1) \subseteq L(r_2) \subseteq \cdots$ collapses iff $\eta^{-1}(P)$ is of star height at most h. It remains to decide whether this hierarchy collapses.

We construct an h-nested distance desert automaton $\mathcal{A}$ which accepts $\eta^{-1}(P)$. The key property of $\mathcal{A}$ is that for every word $w \in \eta^{-1}(P)$, $\Delta_{\mathcal{A}}(w)$ yields the least integer d such that $w \in L(r_{d+1})$. Consequently, $\mathcal{A}$ is limited iff the hierarchy $L(r_1) \subseteq L(r_2) \subseteq \cdots$ collapses iff $\eta^{-1}(P)$ is of star height at most h.

From Section 6.1 to 6.4, we proceed this idea formally. We cannot exactly proceed as in this informal explanation. The empty word ε causes some technical problems, and we rather examine languages $\eta^{-1}(P) \setminus \varepsilon$ than $\eta^{-1}(P)$.

Moreover, most of our constructions proofs are done by an induction on h. Henceforth, we use a two dimensional hierarchy $r_{d,h}$ for $d \geq 1$, $h \in \mathbb{N}$ instead of string expressions $r_1, r_2, \ldots$

In order to examine $\eta^{-1}(P) \setminus \varepsilon$, we construct the expressions $r_{d,h}$ by an induction on h. However, to construct the expression $r_{d,h}$ for some $h \geq 1$, we need the corresponding expressions $r_{d,h-1}$ concerning almost every subset $R \subseteq M$. Thus, we cannot consider $\eta^{-1}(P) \setminus \varepsilon$ separately, we have to consider all the languages $\eta^{-1}(P) \setminus \varepsilon$ for every $P \subseteq M$ in an simultaneous induction. Consequently, our expressions carry one more parameter which is a subset of M, i.e., we consider expressions $r_{d,h}(P)$ for $d \geq 1$, $h \in \mathbb{N}$, $P \subseteq M$. However, we will not use expressions, we rather define languages directly which we will denote by $T_{d,h}(P)$.

6.1 Normal Forms of Rational Expressions

The following easy lemma allows to simplify some techniques, below.

Lemma 6.1. *Let $L \subseteq \Sigma^*$ be recognizable. We have $\mathsf{sh}(L) = \mathsf{sh}(L \setminus \varepsilon)$.*

Proof. We show $\mathsf{sh}(L) \leq \mathsf{sh}(L \setminus \varepsilon)$. If $\varepsilon \notin L$, then the claim is obvious. Otherwise, let r be a rational expression such that $L(r) = L \setminus \varepsilon$ and $\mathsf{sh}(r) = \mathsf{sh}(L)$. We have $L = L(r \cup \varepsilon)$, and hence, $\mathsf{sh}(L) \leq \mathsf{sh}(r \cup \varepsilon) = \mathsf{sh}(r) = \mathsf{sh}(L \setminus \varepsilon)$.

To show $\mathsf{sh}(L \setminus \varepsilon) \leq \mathsf{sh}(L)$, we show that we can transform every rational expression r into an expression r' such that $L(r) \setminus \varepsilon = L(r')$ and $\mathsf{sh}(r) = \mathsf{sh}(r')$. We proceed by an induction on r.

If $r = \emptyset$ or $r \in \Sigma^+$, then we set $r' = r$. If $r = \varepsilon$, then we set $r' = \emptyset$.

If $r = r_1 \cup r_2$, then we transform by induction r_1 and r_2 into r_1' and r_2' and set $r' := r_1' \cup r_2'$.

Assume $r = r_1 r_2$. If $\varepsilon \notin L(r)$, then we set $r' := r$. Otherwise, we have $\varepsilon \in L(r_1)$ and $\varepsilon \in L(r_2)$. We transform r_1 and r_2 into r_1' and r_2' and set $r' := r_1' \cup r_2' \cup r_1' r_2'$.

Finally, assume $r = s^*$. We transform s into s' and set $r' := s' s'^*$. □

We recall the notion of a string expression from R.S. Cohen [4]. We define the notions of a string expression, a single string expression and the degree in a simultaneous induction.

Every word $w \in \Sigma^*$ is a *single string expression* of star height $\mathsf{sh}(w) = 0$ and *degree* $\mathsf{dg}(w) := |w|$. Let $n \geq 1$ and $r_1, \ldots, r_n$ be single string expressions. We call $r := r_1 \cup \cdots \cup r_n$ a *string expression* of star height $\mathsf{sh}(r) = \max\{\mathsf{sh}(r_i) \,|\, 1 \leq i \leq n\}$ and *degree* $\mathsf{dg}(r) := \max\{\mathsf{dg}(r_i) \,|\, 1 \leq i \leq n\}$. The empty set $\emptyset$ is a *string expression* of star height $\mathsf{sh}(\emptyset) = 0$ and *degree* $\mathsf{dg}(\emptyset) := 0$.

Let $n \geq 2$, $a_1, \ldots, a_n \in \Sigma$, and $s_1, \ldots, s_{n-1}$ be string expressions. We call the expression $s := a_1 s_1^* a_2 s_2^* \ldots s_{n-1}^* a_n$ a *single string expression* of star height $\mathsf{sh}(s) = 1 + \max\{\mathsf{sh}(s_i) \,|\, 1 \leq i < n\}$ and *degree* $\mathsf{dg}(s) := \max\big(\{n\} \cup \{\mathsf{dg}(s_i) \,|\, 1 \leq i < n\}\big)$.

String expressions define languages because they are particular rational expressions.

Let r and s be single string expressions. We can construct a single string expression t with $L(t) = L(r)L(s)$ and $\mathsf{sh}(t) = \max\{\mathsf{sh}(r), \mathsf{sh}(s)\}$ as follows: if $\mathsf{sh}(r) \geq 1$ and $\mathsf{sh}(s) \geq 1$, then we set $t := r\,\emptyset^* s$. If $\mathsf{sh}(r) = \mathsf{sh}(t) = 0$, then we simply set $t := rs$. Assume $\mathsf{sh}(r) = 0$ and $\mathsf{sh}(s) \geq 1$. If $r = \varepsilon$, then we set $t := s$. If $r \in \Sigma^+$, we can denote $r = a_1 \ldots a_{|r|}$ and set $t := a_1 \emptyset^* a_2 \emptyset^* \ldots \emptyset^* a_{|r|} \emptyset^* s$. If $\mathsf{sh}(r) \geq 1$ and $\mathsf{sh}(s) = 0$, then we proceed in a symmetric way.

Let r, t be single string expressions with $r \neq \varepsilon$ and $t \neq \varepsilon$, and let s be a string expression. We can regard rs^*t as a string expression with $\mathsf{sh}(rs^*t) = \max\big\{\mathsf{sh}(r),\, 1 + \mathsf{sh}(s),\, \mathsf{sh}(t)\big\}$. If r (or similarly t) is just a word $a_1 \ldots a_{|r|}$, then we understand rs^*t as $a_1 \emptyset^* a_2 \emptyset^* \ldots \emptyset^* a_{|r|} s^* t$.

The following lemma is due to R.S. Cohen [4].

Lemma 6.2. *Let $L \subseteq \Sigma^*$ be a recognizable language. There is a string expression s such that we have $L = L(s)$ and $\mathsf{sh}(s) = \mathsf{sh}(L)$.*

Proof. The lemma is an immediate conclusion from the following claim: for every rational expression r, there is a string expression s such that $L(s) = L(r)$ and $\mathsf{sh}(s) \leq \mathsf{sh}(r)$. To prove this claim, let r be a rational expression.

If r is just a word or $r = \emptyset$, then we let $s := r$.

Assume $r = r'r''$. If $L(r') = \emptyset$ or $L(r'') = \emptyset$, then let $s := \emptyset$. Otherwise, we assume by induction string expressions s' and s'' with $L(s') = L(r')$, $\mathsf{sh}(s') \leq \mathsf{sh}(r') \leq \mathsf{sh}(r)$ and similarly for s''. We denote $s' = s_1' \cup \cdots \cup s_{n'}'$ and $s'' = s_1'' \cup \cdots \cup s_{n''}''$ for suitable n', n'', and single string expressions

$s'_1, \ldots, s'_{n'}, s''_1, \ldots, s''_{n''}$. As seen above, we can concatenate every s'_i and s''_j to a single string expression $s'_i s''_j$ such that $L(s'_i)L(s''_j) = L(s'_i s''_j)$. We set

$$s := \bigcup_{1 \le i \le n',\, 1 \le j \le n''} s'_i s''_j.$$

Clearly, $L(s) = L(r)$ and $\mathsf{sh}(s) \le \mathsf{sh}(r)$.

The case $r = r' \cup r''$ is similar but simpler.

Assume $r = r'^*$. If $L(r') = \emptyset$, then we set $s := \varepsilon$. Assume $L(r') \ne \emptyset$. By induction, let s' be a string expression with $L(s') = L(r')$ and $\mathsf{sh}(s') \le \mathsf{sh}(r') = \mathsf{sh}(r) - 1$. We can denote $s' = s'_1 \cup \cdots \cup s'_{n'}$ for suitable n', $s'_1, \ldots, s'_{n'}$. We set

$$s \;:=\; \varepsilon \;\cup\; s' \;\cup \bigcup_{1 \le i,j \le n',\, s'_i \ne \varepsilon,\, s'_j \ne \varepsilon} s'_i\, s'^*\, s'_j.$$

Clearly, $L(s) = L(r)$ and $\mathsf{sh}(s) \le \mathsf{sh}(r)$. □

6.2 The $T_{d,h}(P)$-hierarchy

We extend η to $\eta : \mathcal{P}(\Sigma^*) \to \mathcal{P}(M)$ as usual. For every $P, R \subseteq M$, we define $R^{-1}P := \{p \in M \mid Rp \subseteq P\}$. For every $P, R \subseteq M$, we have by the definition $R(R^{-1}P) \subseteq P$. For every $P \subseteq M$, the set $P^{-1}P$ is a submonoid of M which is called the *right stabilizer of* P.

Let $d \ge 1$ and $P \subseteq M$. We define $T_{1,0}(P) := \{a \in \Sigma \mid \eta(a) \in P\}$ and for $d > 1$

$$T_{d,0}(P) := \bigcup_{\substack{\text{For every } 1 \le c \le d \text{ and} \\ P_0,\ldots,P_c \subseteq M,\, P_0 = \{1\},\, P_c \subseteq P.}} T_{1,0}\big(P_0^{-1}P_1\big)\, T_{1,0}\big(P_1^{-1}P_2\big) \cdots T_{1,0}\big(P_{c-1}^{-1}P_c\big).$$

Lemma 6.3. *Let $d \ge 1$ and $P \subseteq M$. We have $T_{d,0}(P) = \{w \mid \eta(w) \in P,\ 1 \le |w| \le d\}$.*

Proof. Let $w \in \Sigma^*$ such that $\eta(w) \in P$ and $1 \le |w| \le d$. Denote $w = a_1 \ldots a_{|w|}$ and set $c := |w|$, $P_0 := \{1\}$, and $P_i := \{\eta(a_1 \ldots a_i)\}$ for $1 \le i \le c$. Then, we have for every $1 \le i \le c$, $P_{i-1} \cdot \eta(a_i) = P_i$, i.e., $\eta(a_i) \in P_{i-1}^{-1}P_i$, and hence, $a_i \in T_{1,0}(P_{i-1}^{-1}P_i)$. Consequently, $w \in T_{d,h}(P)$.

Conversely, let $w \in T_{d,0}(P)$. Clearly, $1 \le |w| \le d$. Let $c := |w|$. There are $P_0, \ldots, P_c$ as in the definition of $T_{d,0}(P)$ and $a_i \in T_{1,0}(P_{i-1}^{-1}P_i)$ for $1 \le i \le c$ such that $w = a_1 \ldots a_c$. We have $\eta(a_1) \in P_1$. Let $1 < i \le c$ and assume by induction $\eta(a_1 \ldots a_{i-1}) \in P_i$. Since $\eta(a_i) \in P_{i-1}^{-1}P_i$, we have $P_{i-1} \cdot \eta(a_i) \subseteq P_i$, and hence, $\eta(a_1 \ldots a_i) \in P_i$. Hence, $\eta(w) \in P_c \subseteq P$. □

By Lemma 6.3, we have $\bigcup_{d \ge 1} T_{d,0}(P) = \eta^{-1}(P) \setminus \varepsilon$.

Let $h \in \mathbb{N}$, and assume by induction that for every $P \subseteq M$, $T_{d,h}(P)$ is defined. We define

$$T_{d,\,h+1}(P) := \bigcup_{\substack{\text{For every } 1 \le c \le d \text{ and} \\ P_0,\ldots,P_c \subseteq M,\, P_0 = \{1\},\, P_c \subseteq P.}} T_{1,0}(P_0^{-1}P_1)\Big(T_{d,h}(P_1^{-1}P_1)\Big)^* T_{1,0}(P_1^{-1}P_2)\Big(T_{d,h}(P_2^{-1}P_2)\Big)^* \cdots T_{1,0}(P_c^{-1}P_c).$$

Let $d \ge 1$, $h \in \mathbb{N}$ and $P \subseteq M$ be arbitrary.

From the definition, it follows immediately for every $P \subseteq P' \subseteq M$, $T_{d,h}(P) \subseteq T_{d,h}(P')$.

It is easy to show by an induction on h that for every $d \le d'$, we have $T_{d,h}(P) \subseteq T_{d',h}(P)$.

Since $\varepsilon \in (T_{d,0}(R^{-1}R))^*$ for every $R \subseteq M$, we obtain $T_{d,0}(P) \subseteq T_{d,1}(P)$ from the definition of $T_{d,1}(P)$. Then, we obtain by an induction on h, $T_{d,h}(P) \subseteq T_{d,h+1}(P)$, and for every $h \le h'$, we get $T_{d,h}(P) \subseteq T_{d,h'}(P)$.

To sum up, for every $1 \le d \le d'$, $0 \le h \le h'$, and $P \subseteq P' \subseteq M$, we have $T_{d,h}(P) \subseteq T_{d',h'}(P')$.

Whenever we use the notion $T_{d,h}(P)$-*hierarchy*, we regard $P \subseteq M$ and $h \in \mathbb{N}$ as fixed, i.e., it is a one-dimensional hierarchy w.r.t. the parameter $d \ge 1$.

By induction, we can easily construct a string expression r with $L(r) = T_{d,h}(P)$ such that $\mathsf{sh}(r) \le h$ and $\mathsf{dg}(r) \le d$, and hence, $\mathsf{sh}\big(T_{d,h}(P)\big) \le h$. However, we cannot assume that there is a string expression r with $L(r) = T_{d,h}(P)$ such that $\mathsf{sh}(r) = h$ and $\mathsf{dg}(r) = d$. In the inductive construction of r, several sets $T_{1,0}(P_i^{-1}P_i)$ may be empty, and then, the star-height (resp. degree) of r is possibly smaller than h (resp. d). Just consider the case $T_{d,h}(P) = \{a\}$ but $h > 1$, $d > 1$.

Lemma 6.4. *Let $d \ge 1$, $h \in \mathbb{N}$, and $P \subseteq M$. We have*

$$\big(T_{d,h}(P^{-1}P)\big)^* T_{1,0}(P^{-1}P)\big(T_{d,h}(P^{-1}P)\big)^* \;\subseteq\; \big(T_{d,h}(P^{-1}P)\big)^*.$$

Proof. As seen above, we have $T_{1,0}(P^{-1}P) \subseteq T_{d,h}(P^{-1}P)$. □

Lemma 6.5. *Let $d \ge 1$, $h \in \mathbb{N}$, and $P \subseteq M$. We have $T_{d,h}(P) \subseteq \eta^{-1}(P) \setminus \varepsilon$.*

Proof. We fix some arbitrary $d \ge 1$ for the entire proof.

For $h = 0$, the claim follows from Lemma 6.3.

Let $h \in \mathbb{N}$ and assume by induction that the claim is true for h. For every subset $R \subseteq M$, we have by induction $T_{d,h}(R) \subseteq \eta^{-1}(R) \setminus \varepsilon$. If R is a submonoid, then we have $\big(T_{d,h}(R)\big)^* \subseteq \eta^{-1}(R)$. For every $S \subseteq M$, $S^{-1}S$ is a submonoid of M, and hence, $\big(T_{d,h}(S^{-1}S)\big)^* \subseteq \eta^{-1}(S^{-1}S)$.

Let $P \subseteq M$. We prove the claim for P, d, and $h+1$, i.e., we show $T_{d,h+1}(P) \subseteq \eta^{-1}(P) \setminus \varepsilon$. Let $w \in T_{d,h+1}(P)$ be arbitrary. We have $w \ne \varepsilon$. It remains to show $\eta(w) \in P$. We factorize w according to the definition of $T_{d,h+1}(P)$. There are some $1 \le c \le d$, subsets $P_0, \dots, P_c \subseteq M$, $P_0 = \{1\}$, $P_c \subseteq P$, and there are $a_1, \dots, a_c \in \Sigma$ and $w_1, \dots, w_{c-1} \in \Sigma^*$ such that $w = a_1 w_1 a_2 w_2 \dots w_{c-1} a_c$ and

1. for every $1 \le i \le c$, we have $a_i \in T_{1,0}(P_{i-1}^{-1}P_i)$, i.e., $P_{i-1} \cdot \eta(a_i) \subseteq P_i$ and
2. for every $1 \le i < c$, we have $w_i \in \big(T_{d,h}(P_i^{-1}P_i)\big)^* \subseteq \eta^{-1}(P_i^{-1}P_i)$, and hence, $P_i \cdot \eta(w_i) \subseteq P_i$.

By an induction on i, we have for $1 \le i < c$, $\eta(a_1 w_1 \dots a_i w_i) \in P_i$, and hence, $\eta(w) \in P_c \subseteq P$. □

By Lemma 6.3 and 6.5, we have for every $h \in \mathbb{N}$ and $P \subseteq M$:

$$\eta^{-1}(P) \setminus \varepsilon \;=\; \bigcup_{d \ge 1} T_{d,0}(P) \;\subseteq\; \bigcup_{d \ge 1} T_{d,h}(P) \;\subseteq\; \eta^{-1}(P) \setminus \varepsilon.$$

Let $h \in \mathbb{N}$. We say that the $T_{d,h}(P)$-hierarchy *collapses for* h if there is some $d \ge 1$ such that $T_{d,h}(P) = \eta^{-1}(P) \setminus \varepsilon$. The key question is: for which $h \in \mathbb{N}$ does the $T_{d,h}(P)$-hierarchy collapse? If the $T_{d,h}(P)$-hierarchy collapses for some h, then it collapses for every $h' \ge h$. Hence, we can raise the key question as follows: given some $P \subseteq M$, what is the least h for which the $T_{d,h}(P)$-hierarchy collapses?

Let us consider the particular case $h = 0$. For every $d \ge 1$, the set $T_{d,0}(P)$ is finite. Thus, the $T_{d,0}(P)$-hierarchy collapses iff $\eta^{-1}(P)$ is finite. Consequently, the $T_{d,0}(P)$-hierarchy collapses iff $\eta^{-1}(P)$ is of star height 0. This observation leads us to the guess that the $T_{d,h}(P)$-hierarchy collapses for some $h \in \mathbb{N}$ iff $h \ge \mathsf{sh}\big(\eta^{-1}(P) \setminus \varepsilon\big)$. Below, Lemma 6.6 allows us to prove that our guess is right.

Lemma 6.6. *Let r be a string expression and assume $\varepsilon \notin L(r)$. Let $d \ge \mathsf{dg}(r)$, $h \ge \mathsf{sh}(r)$, and $\eta(L(r)) \subseteq P \subseteq M$. We have $L(r) \subseteq T_{d,h}(P)$.*

Proof. For $r = \emptyset$, the claim is obvious. We assume $r \neq \emptyset$ in the rest of the proof. We proceed by an induction on the star height of r.

Assume $\mathsf{sh}(r) = 0$. Let d, h, and P as in the lemma. There are some $k \geq 1$ and $w_1, \ldots, w_k \in \Sigma^+$ such that $r = w_1 \cup \cdots \cup w_k$ and for every $1 \leq i \leq k$, we have $1 \leq |w_i| \leq d$, and moreover, $\eta(w_i) \in P$. By Lemma 6.3, we have $w_i \in T_{d,0}(P)$, i.e., $L(r) \subseteq T_{d,0}(P) \subseteq T_{d,h}(P)$ for every $h \geq 0$.

Now, let r be a single string expression such that $\mathsf{sh}(r) \geq 1$ and assume that the claim is true for every string expression r' with $\mathsf{sh}(r') < \mathsf{sh}(r)$. We want to show the claim for r. Let $d \geq \mathsf{dg}(r)$ and $h \geq \mathsf{sh}(r)$, and let $\eta(L(r)) \subseteq P \subseteq M$.

Let $1 \leq c \leq d$, and let $a_1, \ldots, a_c \in \Sigma$ and $r_1, \ldots, r_{c-1}$ be string expressions of a star height less than $\mathsf{sh}(r)$ such that $r = a_1 r_1^* a_2 r_2^* \ldots r_{c-1}^* a_c$. We freely assume $\varepsilon \notin L(r_i)$ for every $1 \leq i < c$.

For $1 \leq i < c$, let $P_i := \eta\big(L(a_1 r_1^* \ldots a_i r_i^*)\big)$ and let $P_0 := \{1\}$, $P_c := \eta(L(r))$.

We show $L(r) \subseteq T_{d,h}(P)$, by applying the definition of $T_{d,h}(P)$ using $P_0, \ldots, P_c$. To complete the proof for r, we consider the following two assertions:

1. for every $1 \leq i \leq c$, we have $a_i \in T_{1,0}(P_{i-1}^{-1} P_i)$, and

2. for every $1 \leq i < c$, we have $L(r_i) \subseteq T_{d,h-1}(P_i^{-1} P_i)$.

(1) Let $1 \leq i \leq c$. By the definition of $P_0, \ldots, P_c$, we have $P_{i-1} \cdot \eta(L(a_i r_i^*)) = P_i$, and since $a_i \in L(a_i r_i^*)$, we have $P_{i-1} \cdot \eta(a_i) \subseteq P_i$, i.e., $\eta(a_i) \in P_{i-1}^{-1} P_i$. Then, (1) follows from the definition of $T_{1,0}(P_{i-1}^{-1} P_i)$.

(2) Let $1 \leq i < c$. Since $L(a_1 r_1^* \ldots a_i r_i^*) L(r_i) \subseteq L(a_1 r_1^* \ldots a_i r_i^*)$, we have $P_i \cdot \eta(L(r_i)) \subseteq P_i$, and hence, $\eta(L(r_i)) \subseteq P_i^{-1} P_i$. We have $d \geq \mathsf{dg}(r) \geq \mathsf{dg}(r_i)$, and since $h \geq \mathsf{sh}(r) > \mathsf{sh}(r_i)$, we have $h - 1 \geq \mathsf{sh}(r_i)$. Thus, we can apply the inductive hypothesis and obtain $L(r_i) \subseteq T_{d,h-1}(P_i^{-1} P_i)$.

This completes the proof for r.

Finally, let r be a string expression such that $\mathsf{sh}(r) \geq 1$. Let $d \geq \mathsf{dg}(r)$ and $h \geq \mathsf{sh}(r)$, and let $\eta(L(r)) \subseteq P \subseteq M$. There are some $k \geq 1$ and single string expressions $r_1, \ldots, r_k$ such that $r = r_1 \cup \cdots \cup r_k$. For every $1 \leq i \leq k$, we have $d \geq \mathsf{dg}(r_i)$, $h \geq \mathsf{sh}(r_i)$, and $\eta(L(r_i)) \subseteq P$. Since we have already shown the claim for single string expressions, we have $L(r_i) \subseteq T_{d,h}(P)$, i.e., $L(r) \subseteq T_{d,h}(P)$. □

Proposition 6.7. *Let $h \in \mathbb{N}$ and $P \subseteq M$. There is a $d \geq 1$ such that $T_{d,h}(P) = \eta^{-1}(P) \setminus \varepsilon$ iff $\mathsf{sh}\big(\eta^{-1}(P)\big) \leq h$.*

Proof. $\Rightarrow$ As already mentioned, we have $h \geq \mathsf{sh}(T_{d,h}(P)) = \mathsf{sh}(\eta^{-1}(P) \setminus \varepsilon)$, and by Lemma 6.1, we have $\mathsf{sh}(\eta^{-1}(P) \setminus \varepsilon) = \mathsf{sh}(\eta^{-1}(P))$.

$\Leftarrow$ By Lemma 6.1, we have $\mathsf{sh}(\eta^{-1}(P) \setminus \varepsilon) \leq h$. By Lemma 6.2, there is a string expression r with $L(r) = \eta^{-1}(P) \setminus \varepsilon$ and $\mathsf{sh}(r) \leq h$. Let $d := \mathsf{dg}(r)$. We have $\eta(L(r)) \subseteq P$. By Lemma 6.6 and 6.5, we obtain $\eta^{-1}(P) \setminus \varepsilon = L(r) \subseteq T_{d,h}(P) \subseteq \eta^{-1}(P) \setminus \varepsilon$. □

6.3 A Reduction to Limitedness

In this section, we construct for given $h \in \mathbb{N}$ and $P \subseteq M$ a $(h+1)$-nested distance desert automaton $\mathcal{A}_h(P)$ which computes on words w the least level of the $T_{d,h}(P)$-hierarchy which contains w. Consequently, $\mathcal{A}_h(P)$ is limited iff the $T_{d,h}(P)$-hierarchy collapses. In combination with Proposition 6.7 and the decidability of limitedness of nested distance desert automata, this construction allows to decide whether the star height of the languages $\eta^{-1}(P)$ is less than h.

Proposition 6.8. *Let $h \in \mathbb{N}$ and $P \subseteq M$. We can construct a $(h+1)$-nested distance desert automaton $\mathcal{A}_h(P) = [Q, E, q_I, q_F, \theta]$ with the following properties:*

1. $E \subseteq (Q \setminus q_F) \times \Sigma \times (Q \setminus q_I)$,
2. $|Q| \leq 2^{|M|h}\big(|M| + 2\big)$,
3. *for every $(p, a, q) \in E$, we have $\theta((p, a, q)) = \Upsilon_h$ if $p = q_I$, and* $\theta((p, a, q)) \in \{\Upsilon_0, \ldots, \Upsilon_{h-1}, \angle_0, \ldots, \angle_h\}$ *if $p \neq q_I$,*
4. *for every $w \in \Sigma^*$, $\Delta(w) + 1 = \min\{d \geq 1 \mid w \in T_{d,h}(P)\}$.*

As a conclusion from (4), we have $L\big(\mathcal{A}_h(P)\big) = \bigcup_{d \geq 1} T_{d,h}(P) = \eta^{-1}(P) \setminus \varepsilon$. Indeed, for every $w \in L\big(\mathcal{A}_h(P)\big)$, we have $\Delta(w) \in \mathbb{N}$, and by (4), w belongs to some $T_{d,h}(P)$. Conversely, if w belongs to $T_{d,h}(P)$ for some $d \geq 1$ then (4) implies $\Delta(w) \in \mathbb{N}$, i.e., $w \in L\big(\mathcal{A}_h(P)\big)$.

The crucial property of $\mathcal{A}_h(P)$ is that due to (4), $\mathcal{A}_h(P)$ is limited iff there is some $d \geq 1$ such that $T_{d,h}(P) = \eta^{-1}(P) \setminus \varepsilon$.

Proof of Proposition 6.8. For $P = \emptyset$, the construction is straightforward by setting $Q = \{q_I, q_F\}$ and $E = \emptyset$. Then, (1), (2), (3) are obviously satisfied, and (4) is satisfied since the equation comes up to $\infty = \infty$ for every $w \in \Sigma^*$. We assume $P \neq \emptyset$ in the rest of the proof.

Recall that $\mathcal{P}_{ne}(M)$ denotes the set of all non-empty subsets of M.

We proceed by induction on h. Let $P \in \mathcal{P}_{ne}(M)$ be arbitrary.

Let $h = 0$. At first, we construct an automaton which accepts $\eta^{-1}(P)$. We use M as states. For every $q \in M$, $a \in \Sigma$, we set a transition $(q, a, q \cdot \eta(a))$. The initial state is $\{1\}$ and P are the accepting states. We apply to this automaton a standard construction to get an automaton $[Q, E, q_I, q_F]$ which recognizes $\eta^{-1}(P) \setminus \varepsilon$ and satisfies (1) whereas $Q = M \cup \{q_I, q_F\}$. Hence, $|Q| = |M| + 2$, i.e., (2) is satisfied. For every transition $(q_I, a, q) \in E$, we set $\theta((q_I, a, q)) = \Upsilon_0$. For every transition $(p, a, q) \in E$ with $p \neq q_I$, we set $\theta((p, a, q)) = \angle_0$. This completes the construction of $\mathcal{A}_0(P) = [Q, E, q_I, q_F, \theta]$, and (3) is satisfied.

We show (4). For $w = \varepsilon$, (4) comes up to $\infty = \infty$. Let $w \in \Sigma^+$. If $w \notin \eta^{-1}(P)$, then (4) comes up to $\infty = \infty$. Assume $w \in \eta^{-1}(P)$. Denote $w = a_1 \ldots a_{|w|}$. Then, $\mathcal{A}_0(P)$ accepts w. By the definition of θ, it is quite clear that every successful path for w has the weight $|w| - 1$, i.e., $\Delta(w) = |w| - 1$. By the definition of $T_{d,0}(P)$, $d := |w|$ is the least integer for which $w \in T_{d,0}(P)$. Hence, (4) comes up to $|w| = |w|$.

Now, let $h \in \mathbb{N}$. We assume by induction that the claim is true for h for every $P \subseteq M$.

Let $P \in \mathcal{P}_{ne}(M)$ be arbitrary. We show the claim for $h+1$ and P. At first, we construct an automaton $\mathcal{A}' := [Q', E', q_I, q_F, \theta']$. Let $Q' := \mathcal{P}_{ne}(M) \cup \{q_I, q_F\}$.

Let $a \in \Sigma$ and $S, T \in \mathcal{P}_{ne}(M)$ be arbitrary. If $S \neq T$ and $S \cdot \eta(a) \subseteq T$, then we put the transition (S, a, T) into E'. If $\eta(a) \in T$, then we put (q_I, a, T) into E'. If $S \cdot \eta(a) \subseteq P$, then we put the transition (S, a, q_F) into E'. Finally, if $\eta(a) \in P$, then we put (q_I', a, q_F') into E'.

For every word w which $\mathcal{A}'$ accepts, we have $\eta(w) \in P$. However, we cannot show that $\mathcal{A}'$ accepts every non-empty word in $\eta^{-1}(P)$ since we did not allow self loops (S, a, S) even if $S \cdot \eta(a) \subseteq S$.

We define $\theta' : E' \to \{\Upsilon_{h+1}, \angle_{h+1}\}$. For every transition $(q_I, a, q) \in E'$, let $\theta'((q_I, a, q)) = \Upsilon_{h+1}$. For every transition $(p, a, q) \in E'$ with $p \neq q_I$, we set $\theta'((p, a, q)) = \angle_{h+1}$.

We construct $\mathcal{A}_{h+1}(P)$. By the inductive hypothesis, we can construct an automaton $\mathcal{A}_h(S^{-1}S)$ which satisfies (1, ..., 4). We insert into $\mathcal{A}'$ a disjoint copy of $\mathcal{A}_h(S^{-1}S)$, and we identify both the initial and accepting state of $\mathcal{A}_h(S^{-1}S)$ with the state S in $\mathcal{A}'$. We proceed this insertion for every

$S \in \mathcal{P}_{ne}(M)$, i.e., for every state in $\mathcal{A}'$ except the initial and accepting state. In this way, we achieve an automaton $[Q, E, q_I, q_F]$. In this construction, the union of the transitions is disjoint since we did not allow self loops in $\mathcal{A}'$. Hence, it arises a mapping θ as a union of θ' and the corresponding mappings of the automata $\mathcal{A}_h(S^{-1}S)$. We obtain the automaton $\mathcal{A}_{h+1}(P) = [Q, E, q_I, q_F, \theta]$.

Let $t \in E$. If $\theta(t) \in \{\angle_{h+1}, \Upsilon_{h+1}\}$, then t stems from $\mathcal{A}'$. If $\theta(t) \in \{\Upsilon_0, \ldots, \Upsilon_h, \angle_0, \ldots, \angle_h\}$, then t stems from a one of the inserted automata $\mathcal{A}_h(S^{-1}S)$.

Clearly, $\mathcal{A}_{h+1}(P)$ satisfies (1) and (3). We show (2). Every inserted copy of some automaton $\mathcal{A}_h(S^{-1}S)$ has at most $2^{|M|h}\big(|M|+2\big)-1$ states since we lose one state in the identification of the initial and accepting state. We insert $|\mathcal{P}_{ne}(M)| = 2^{|M|}-1$ copies. Hence,

$$|Q'| \;\leq\; (2^{|M|}-1)\Big(2^{|M|h}\big(|M|+2\big)-1\Big)+2 \;<\; 2^{|M|}\Big(2^{|M|h}\big(|M|+2\big)-1\Big)+2 \;=\; \ldots$$

$$\ldots \;=\; 2^{|M|(h+1)}\big(|M|+2\big)-2^{|M|}+2 \;\leq\; 2^{|M|(h+1)}\big(|M|+2\big),$$

i.e., (2) is satisfied.

We show (4). For $w = \varepsilon$ the equation in (4) comes up to $\infty = \infty$. Let $w \in \Sigma^+$ be arbitrary. To prove (4) for w, we show the following two claims:

4a. Let $d \geq 1$. For every $w \in T_{d,h+1}(P)$, there is a successful path π in $\mathcal{A}_{h+1}(P)$ with the label w and $\Delta(\theta(\pi))+1 \;\leq\; d$.

4b. Let π be a successful path in $\mathcal{A}_{h+1}(P)$ with the label w. We have $w \in T_{\Delta(\theta(\pi))+1,\,h+1}(P)$.

Out next step is to show that (4a) and (4b) together prove (4).

Let $w \in \Sigma^+$. We want to show

$$\Delta(w)+1 \;\leq\; \mathsf{min}\big\{d \geq 1 \,\big|\, w \in T_{d,h}(P)\big\}$$

in (4). If $\eta(w) \in P$, then let d be the least integer such that $w \in T_{d,h+1}(P)$. By (4a), we have $\Delta(w)+1 \;\leq\; d$. If $\eta(w) \notin P$, then there is no integer d such that $w \in T_{d,h+1}(P)$. Then, the minimum yields ∞ and we always have $\Delta(w)+1 \;\leq\; \infty$.

Let $w \in \Sigma^+$. To complete the proof of (4) from (4a) and (4b), we show

$$\Delta(w)+1 \;\geq\; \mathsf{min}\big\{d \geq 1 \,\big|\, w \in T_{d,h}(P)\big\}.$$

Assume that $\mathcal{A}_{h+1}(P)$ accepts w. Let π be an successful path in $\mathcal{A}_{h+1}(P)$ with the label w such that $\Delta(\theta(\pi)) = \Delta(w)$. By (4b), we have $w \in T_{\Delta(w)+1,\,h+1}(P)$, i.e., the minimum yields at most $\Delta(w)+1$. If $\mathcal{A}_{h+1}(P)$ does not accept w, then $\Delta(w)+1 = \infty$, and clearly, the minimum cannot yield a result larger than ∞.

Consequently, (4a) and (4b) together prove (4), but it remains to prove (4a) and (4b).

We show (4a). Let $d \geq 1$ and $w \in T_{d,h+1}(P)$. We factorize w according to the definition of $T_{d,h+1}(P)$. There are some $1 \leq c \leq d$ and $P_0, \ldots, P_c \subseteq M$, $P_0 = \{1\}$, $P_c \subseteq P$. For $1 \leq i \leq c$, let $R_i := P_i^{-1}P_i$. For every $1 \leq i \leq c$, there is some $a_i \in T_{1,0}(P_{i-1}^{-1}P_i)$, and for every $1 \leq i < c$ there is some $w_i \in \big(T_{d,h}(R_i)\big)^*$ such that $w = a_1 w_1 a_2 w_2 \ldots a_c$.

By contradiction, assume $P_i = \emptyset$ for some $1 \leq i \leq c$. Let i be the least integer such that $P_i = \emptyset$. Hence $P_{i-1}^{-1}P_i = \emptyset$ and $T_{1,0}(P_{i-1}^{-1}P_i) = \emptyset$ which contradicts $a_i \in T_{1,0}(P_{i-1}^{-1}P_i)$.

If $c = 1$, then w is just a letter. We set $\pi := (q_I, w, q_F)$. Then, $\theta(\pi) = \Upsilon_{h+1}$ and $\Delta(\theta(\pi)) = 0$ which proves (4a). We assume $c \geq 2$ in the rest of the proof of (4a).

Assume that there is some $1 \le i < c-1$ such that $P_i = P_{i+1}$. By Lemma 6.4, we have

$$\big(T_{d,h}(R_i)\big)^* T_{1,0}(P_i^{-1}P_{i+1})\big(T_{d,h}(R_{i+1})\big)^* \;\subseteq\; \big(T_{d,h}(R_i)\big)^*,$$

and hence, $w_i a_{i+1} w_{i+1} \in \big(T_{d,h}(R_i)\big)^*$. Consequently, we can factorize w according to the definition of $T_{d,h+1}(P)$ using the sets $P_1, P_2, \dots, P_i, P_{i+2}, \dots, P_c$. By applying this argument as many times as possible, we can assume $P_i \ne P_{i+1}$ for $1 \le i < c-1$. However, we allow $P_{c-1} = P_c$.

We construct a path π to prove (4a). Let $t_1 := (q_I, a_1, P_1)$ and $t_c := \big(P_{c-1}, a_c, q_F\big)$. For every $2 \le i < c$, let $t_i := \big(P_{i-1}, a_i, P_i\big)$. Clearly, $t_1, \dots, t_c$ are transitions in $\mathcal{A}_{h+1}(P)$, $\theta(t_1) = \curlyvee_{h+1}$, and for $2 \le i \le c$, $\theta(t_i) = \angle_{h+1}$.

Let $1 \le i < c$. We factorize w_i. There are some $n_i \in \mathbb{N}$ and $w_{i,1}, \dots, w_{i,n_i} \in T_{d,h}(R_i)$ such that $w_i = w_{i,1} \dots w_{i,n_i}$.

Let $1 \le i < c$ and $1 \le j \le n_i$. Then, $w_{i,j} \in T_{d,h}(R_i)$. By the inductive hypothesis, there is an successful path $\tilde{\pi}_{i,j}$ in $\mathcal{A}_h(R_i)$ with the label $w_{i,j}$, $\theta(\tilde{\pi}_{i,j}) \in \curlyvee_h\{\curlyvee_0, \dots, \curlyvee_{h-1}, \angle_0, \dots, \angle_h\}^*$, and $\Delta(\theta(\tilde{\pi}_{i,j})) + 1 \le d$. Since there is a copy of $\mathcal{A}_h(R_i)$ in $\mathcal{A}_{h+1}(P)$ at the state P_i, there is a copy of $\tilde{\pi}_{i,j}$ in $\mathcal{A}_{h+1}(P)$ from P_i to P_i. Let $\pi_{i,j}$ be the copy of $\tilde{\pi}_{i,j}$ in $\mathcal{A}_{h+1}(P)$. The path $\pi_{i,j}$ starts and ends at P_i and has the same label and mark as $\tilde{\pi}_{i,j}$.

For every $1 \le i < c$, let $\pi_i := \pi_{i,1} \dots \pi_{i,n_i}$. If $n_i = 0$, then π_i is simply the empty path from P_i to P_i. The path π_i is labeled with w_i.

Let $\pi = t_1\pi_1t_2\pi_2 \cdots t_c$. Clearly, π is an successful path in $\mathcal{A}_{h+1}(P)$ and π is labeled with $a_1w_1a_2w_2 \dots a_c = w$. To show (4a), it remains to show $\Delta(\theta(\pi)) + 1 \le d$. Let π' be an arbitrary factor of $\theta(\pi)$. We have $|\pi'|_{h+1} + 1 \le c \le d$. Let $0 \le g \le h$, and assume that $\pi' \in \{\curlyvee_0, \dots, \curlyvee_{g-1}, \angle_0, \dots, \angle_g\}^*$, i.e., we cannot obtain g-coins in π'. Then, π' stems from some path $\pi_{i,j}$, i.e., from some path $\tilde{\pi}_{i,j}$ in $\mathcal{A}_h(R_i)$. Then, $\Delta(\pi') + 1 \le \Delta(\theta(\pi_{i,j})) \le d$ by the choice of $\tilde{\pi}_{i,j}$. Hence, $\Delta(\theta(\pi)) + 1 \le d$.

We show (4b). Let π be a successful path in $\mathcal{A}_{h+1}(P)$ with the label w. Denote $d := \Delta(\theta(\pi)) + 1$.

If $|w| = 1$, then we have $\eta(w) \in P$ by the construction of $\mathcal{A}_{h+1}(P)$. Hence, $w \in T_{1,h+1}(P)$, and we are done. We assume $|w| \ge 1$ in the rest of the proof of (4b).

The first transition of π is marked by $\curlyvee_{h+1}$, any other transitions are marked by some member of $\{\curlyvee_0, \dots, \curlyvee_h, \angle_0, \dots, \angle_{h+1}\}$. Let $c > 1$ and factorize π into $\pi = t_1\pi_1t_2\pi_2 \dots t_c$ such that $t_2, \dots, t_c$ are the transitions in π which are marked by $\angle_{h+1}$. We have $c - 1 \le \Delta(\theta(\pi))$, i.e., $c \le d$.

We denote the labels of $t_1, \dots, t_c$ and $\pi_1, \dots, \pi_{c-1}$ by $a_1, \dots, a_c$ and $w_1, \dots, w_{c-1}$, respectively. Hence, $w = a_1w_1a_2w_2 \dots a_c$. The transitions $t_1, \dots, t_c$ belong to $\mathcal{A}'$. Thus, every transition $t_1, \dots, t_c$ starts and ends at some state in $\mathcal{P}_{ne}(M)$ except t_1 which starts in q_I and t_c which ends in q_F.

Let $1 \le i < c$. We consider the factor $t_i\pi_it_{i+1}$ of π. The path π_i starts and ends in some state in $\mathcal{P}_{ne}(M)$. Since the transitions of π_i are marked by members of $\{\curlyvee_0, \dots, \curlyvee_h, \angle_0, \dots, \angle_h\}$, π_i is in one and the same copy of some automaton $\mathcal{A}_h(S^{-1}S)$. Thus, π_i starts and ends in the same state, i.e., t_{i+1} starts in the state in which t_i ends. Hence, $t_1 \dots t_c$ is an successful path in $\mathcal{A}'$.

For $1 \le i < c$, let P_i be the state in which π_i starts and ends, and let $P_0 = \{1\}$ and $P_c = P$. We use the states $P_0, \dots, P_c$ to show that $w \in T_{d,\,h+1}(P)$.

Let $1 < i < c$. Since $t_i = (P_{i-1}, a_i, P_i)$ is a transition of $\mathcal{A}'$, we have $P_{i-1} \cdot \eta(a_i) \subseteq P_i$, i.e., $\eta(a_i) \in P_{i-1}^{-1}P_i$, and hence, $a_i \in T_{1,0}(P_{i-1}^{-1}P_i)$. Similarly, $a_1 \in T_{1,0}(P_0^{-1}P_1)$ and $a_c \in T_{d,h}(P_{c-1}^{-1}, P_c)$.

It remains to show $w_i \in \big(T_{d,h}(P_i^{-1}P_i)\big)^*$ for $1 \le i < c$. Let $1 \le i < c$. We denote $R_i := P_i^{-1}P_i$. There are some $n_i \in \mathbb{N}$ and non-empty paths $\pi_{i,1}, \dots, \pi_{i,n_i}$ such that $\pi_i = \pi_{i,1} \dots \pi_{i,n_i}$, each of the paths $\pi_{i,1}, \dots, \pi_{i,n_i}$ starts and ends in P_i, and none of the paths $\pi_{i,1}, \dots, \pi_{i,n_i}$ contains the state P_i inside.

Let $1 \leq j \leq n_i$. We denote the label of $\pi_{i,j}$ by $w_{i,j}$, i.e., $w_i = w_{i,1} \ldots w_{i,n_i}$. By renaming the first and last state of $\pi_{i,j}$ to q_I and q_F, resp., we obtain an successful path $\tilde{\pi}_{i,j}$ in $\mathcal{A}_h(R_i)$. The path $\tilde{\pi}_{i,j}$ has the same label and mark like $\pi_{i,j}$. Since $\pi_{i,j}$ is a factor of π, we have $d = \Delta(\theta(\pi)) + 1 \geq \Delta(\theta(\pi_{i,j})) + 1 = \Delta(\theta(\tilde{\pi}_{i,j})) + 1$. Hence, we can apply the inductive hypothesis (claim (4) for $w_{i,j}$ and $\mathcal{A}_h(R_i)$) and obtain $w_{i,j} \in T_{d,h}(R_i)$. Consequently, $w_i \in \big(T_{d,h}(R_i)\big)^*$. □

Let $P \subseteq M$, $h \in \mathbb{N}$, and assume $\mathsf{sh}\big(\eta^{-1}(P)\big) > h$. By Proposition 6.7, the $T_{d,h}(P)$-hierarchy does not collapse. Hence, the automaton $\mathcal{A}_h(P)$ from Proposition 6.8 is not limited. By Theorem 2.1, there is a $\sharp$-expression r such that $\mathcal{A}_h(P)$ accepts $r(k)$ for every $k \geq 1$, but for increasing integers k the weight of $r(k)$ is unbounded. Let $r(\mathbb{N}) := \{r(k) \,|\, k \geq 1\}$. Note that $r(\mathbb{N})$ is not necessarily recognizable. By Proposition 6.8(4), we have for every $d \geq 1$, $r(\mathbb{N}) \not\subseteq T_{d,h}(P)$. Thus, the words $r(\mathbb{N})$ are some kind of witnesses which prove that the $T_{d,h}(P)$-hierarchy does not collapse.

Moreover, let K be a recognizable language such that $r(\mathbb{N}) \subseteq K \subseteq \eta^{-1}(P)$. By contradiction, assume that $\mathsf{sh}(K) \leq h$. By Lemma 6.2, there is a string expression s such that $L(s) = K$ and $\mathsf{sh}(s) \leq h$. Let $d := \mathsf{dg}(s)$. Since $L(s) = K \subseteq \eta^{-1}(P)$, we have $\varepsilon \notin L(s)$ and $\eta(L(s)) \subseteq P$. By Lemma 6.6, we have $L(s) \subseteq T_{d,h}(P)$, and hence, $r(\mathbb{N}) \subseteq T_{d,h}(P)$ which is a contradiction.

Consequently, every recognizable language K such that $r(\mathbb{N}) \subseteq K \subseteq \eta^{-1}(P)$ is of a star height larger than h.

6.4 The Decidability of the Star Height Problem

Proof of Theorem 2.3. Let $h \in \mathbb{N}$ and L be accepted by an n-state nondeterministic automaton. By any proof of KLEENE's theorem, we can show $\mathsf{sh}(L) \leq n$.

An algorithm which decides whether $\mathsf{sh}(L) \leq h$ checks at first whether $n \leq h$. If so, then the algorithm answers "yes".

If $n > h$, then the algorithm proceeds as follows: it constructs the syntactic monoid M, the syntactic homomorphism $\eta : \Sigma^* \to M$, and the set $P \subseteq M$ such that $L = \eta^{-1}(P)$. Then, it constructs the automaton $\mathcal{A}_h(P)$ by Proposition 6.8, and it decides by Theorem 2.2 whether $\mathcal{A}_h(P)$ is limited. The algorithm answers "yes" if $\mathcal{A}_h(P)$ is limited, otherwise it answers "no".

By Lemma 6.1, we have $\mathsf{sh}(L) = \mathsf{sh}\big(\eta^{-1}(P) \setminus \varepsilon\big)$. By Proposition 6.7, we have $\mathsf{sh}\big(\eta^{-1}(P) \setminus \varepsilon\big) \leq h$ iff the $T_{d,h}(P)$-hierarchy collapses. By Proposition 6.8(4), the $T_{d,h}(P)$-hierarchy collapses iff $\mathcal{A}_h(P)$ is limited.

Clearly, the initial test whether $n \leq h$ is not necessary for the correctness of the algorithm. However, this test increases the efficiency and it simplifies the analysis of the complexity.

We have $|M| \leq 2^{n^2}$. The number of states of $\mathcal{A}_h(P)$ is at most $2^{(2^{n^2})h}\,(2^{n^2} + 1)$. Since $n > h$, $\mathcal{A}_h(P)$ has at most $2^{(2^{n^2})n + n^2} + 2^{(2^{n^2})n}$ states. By Theorem 2.2, an algorithm requires $2^{2^{\mathcal{O}(n^2)}}$ space to decide whether $\mathcal{A}_h(P)$ is limited. □

6.5 PSPACE-Hardness of the Star Height Problem

In this section, we show that the star height problem over two letter alphabet is PSPACE-hard. Although this result seems to be well-known, the author did not find any proof in the literature. We reduce the star height problem to the universality problem of nondeterministic automata.

Lemma 6.9. *Let $\Sigma = \{a, b\}$ and $K, L \subseteq \Sigma^*$ be recognizable. Define $L' := \Sigma^* cL \cup Kc\Sigma^*$.*

1. *If $L = \Sigma^*$, then $\mathsf{sh}(L') = 1$.*

2. *If $L \subsetneq \Sigma^*$, then* $\mathsf{sh}(L') \geq \mathsf{sh}(K)$.

Proof. (1) If $L = \Sigma^*$, then $L' = \Sigma^* c \Sigma^*$, and hence, $\mathsf{sh}(L') = 1$.

(2) Let $h := \mathsf{sh}(L')$. For $K = \emptyset$, the claim is vacuously true. We assume $K \neq \emptyset$ in the rest of the proof. Hence, L' is infinite and $h \geq 1$.

We construct a rational expression r such that $L(r) = K$ and $\mathsf{sh}(r) \leq h$. By Lemma 6.2, there is a string expression s such that $L(s) = L'$ and $\mathsf{sh}(s) = h$. There are some $n \geq 1$ and single string expressions $s_1, \ldots, s_n$ such that $s = s_1 \cup \cdots \cup s_n$ and $\mathsf{sh}(s_i) \leq h$ for every $1 \leq i \leq n$.

Let $1 \leq i \leq n$ be arbitrary. There are some n_i, letters $a_1, \ldots, a_{n_i} \in \Sigma \cup \{c\}$ and string expressions $t_1, \ldots, t_{n_i - 1}$ such that $s_i = a_1 t_1^* a_2 t_2^* \ldots t_{n_i-1}^* a_{n_i}$. Moreover, we have $\mathsf{sh}(t_j) < h$ for every $1 \leq j < n_i$. Note that the letters $a_1, \ldots, a_{n_i}$ and $t_1, \ldots, t_{n_i-1}$ depend on i.

By contradiction, assume that c occurs in an expression t_j for some $1 \leq j < n_i$. Then, $L(t_j^*)$ and $L(s)$ contain words with more than one occurrence of c. Hence, c cannot occur in t_j. Since every word in L' contains exactly one occurrence of c, exactly one of the letters $a_1, \ldots, a_{n_j}$ is c. Let $1 \leq l \leq n_i$ such that $a_l = c$ and let $r_i := a_1 K_1^* \ldots a_{l-1} K_{l-1}^*$. Note that r_i is not a string expression, because it ends by K_{l-1}^*. If $l = 1$, then $r_i = \varepsilon$. Similarly, let $r_i' := K_l^* a_{l+1} \ldots K_{n_i-1}^* a_{n_i}$, i.e., we have $s_i = r_i c r_i'$. We assume such expressions r_i and r_i' for every $1 \leq i \leq n$. We define

$$r := \bigcup_{1 \leq i \leq n,\ L(r_i) \subseteq K} r_i$$

We have $\mathsf{sh}(r) \leq \mathsf{sh}(s) = h = \mathsf{sh}(L')$. It remains to show $L(r) = K$. From the definition of r, it follows immediately $L(r) \subseteq K$. Let $w \in K$ be arbitrary. We want to show $w \in L(r)$. Let $u \in \Sigma^* \setminus L$ be arbitrary. We have $wcu \in L'$. Hence, there is some $1 \leq i \leq n$ such that $wcu \in L(s_i)$, i.e., $w \in L(r_i)$ and $u \in L(r_i')$. Thus, $L(r_i)cu \subseteq L'$. Since $u \notin L$, we have $L(r_i) \subseteq K$, and in particular, $w \in L(r_i) \subseteq L(r)$. □

Proposition 6.10. *Let $h \geq 1$. To decide whether for a nondeterministic automaton $\mathcal{A}$ over a three-letter alphabet, we have $\mathsf{sh}(L(\mathcal{A})) \leq h$ is PSPACE-hard.*

Proof. Let $h \geq 1$. Let $K \subseteq \{a, b\}^*$ be a recognizable language such that $\mathsf{sh}(K) = h + 1$. Such a language K exists due to [6].

Let $L \subseteq \{a, b\}^*$ be the language of some nondeterministic automaton. To decide whether $L = \Sigma^*$ is PSPACE-complete [20]. We construct an automaton $\mathcal{A}$ such that $L(\mathcal{A}) = \Sigma^* c L \cup K c \Sigma^*$, i.e., $\mathcal{A}$ accepts L' from Lemma 6.9. Note that K does not depend on L, i.e., $\mathcal{A}$ has just a bounded number of states more than the automaton for L. If $L = \Sigma^*$, then we know by Lemma 6.9(1), $\mathsf{sh}(L') = 1$, and in particular $\mathsf{sh}(L') \leq h$. Conversely, if $L \subsetneq \Sigma^*$, then we know by Lemma 6.9(2), $\mathsf{sh}(L') \geq \mathsf{sh}(K) > h$. To sum up, we have $L = \Sigma^*$ iff $\mathsf{sh}(L') \leq h$. Consequently, to decide whether $\mathsf{sh}(L') \leq h$ it is PSPACE-hard. □

In order to generalize Proposition 6.10, we apply a homomorphism which preserves star height. Let Γ and Σ be two alphabets and let $\alpha : \Gamma^* \to \Sigma^*$ be a homomorphism. For every recognizable language $L \subseteq \Gamma^*$, we have $\mathsf{sh}(L) \geq \mathsf{sh}(\alpha(L))$. We say that α *preserves star height* if for every recognizable language $L \subseteq \Gamma^*$, we have $\mathsf{sh}(L) = \mathsf{sh}(\alpha(L))$.

Assume that α is injective. By following [19], we say that α has the *tag property* if for every $u_1, u_2, v_1, v_2 \in \Sigma^*$ with $u_1 v_1,\ u_1 v_2,\ u_2 v_1,\ u_2 v_2 \in \alpha(\Gamma^*)$, we have one of the following conditions:

1. There are $x, u_1', u_2' \in \Sigma^*$ such that $u_1 = u_1' x$, $u_2 = u_2' x$ and $u_1',\ u_2',\ x v_1,\ x v_2 \in \alpha(\Gamma^*)$.

2. There are $y, v_1', v_2' \in \Sigma^*$ such that $v_1 = yv_1',\ v_2 = yv_2'$ and $u_1y,\ u_2y,\ v_1',\ v_2' \in \alpha(\Gamma^*)$.

We use the following theorem by K. Hashiguchi and N. Honda from 1976 [19].

Theorem 6.11. [19] *A homomorphism $\alpha : \Gamma^* \to \Sigma^*$ preserves star height iff α is injective and α has the tag property.*

Lemma 6.12. *Let $\Gamma = \{a, b, c\}$ and $\Sigma = \{a, b\}$. The homomorphism $\alpha : \Gamma^* \to \Sigma^*$ defined by $\alpha(a) := aa$, $\alpha(b) := ab$, and $\alpha(c) := ba$ preserves star height.*

Proof. By Theorem 6.11, it suffices to show that α is injective and α has the tag property. Obviously, α is injective.

We show that α has the tag property. Let $u_1, u_2, v_1, v_2 \in \Sigma^*$ with $u_1v_1, u_1v_2, u_2v_1, u_2v_2 \in \alpha(\Gamma^*)$. We show that one of the two conditions in the definition of the tag property holds.

If $|u_1|$ is even, then $|v_1|$, $|u_2|$, and $|v_2|$ are even, and both conditions hold for $x = \varepsilon$ (resp. $y = \varepsilon$).

In the rest of the proof, we assume that $|u_1|$ is odd. Hence, $|v_1|$, $|u_2|$, and $|v_2|$ are odd. We factorize $u_1 = u_1'x_1$, $u_2 = u_2'x_2$, $v_1 = y_1v_1'$ and $v_2 = y_2v_2'$ whereas $x_1, x_2, y_1, y_2 \in \Sigma$. Clearly, $|u_1'|$, $|u_2'|$, $|v_1'|$, and $|v_2'|$ are even and $u_1', u_2', v_1', v_2' \in \alpha(\Gamma^*)$.

If $x_1 = x_2$, then condition (1) holds for $x := x_1 = x_2$. If $y_1 = y_2$, then condition (2) holds for $y := y_1 = y_2$. It remains to consider the case that $x_1 \neq x_2$ and $y_1 \neq y_2$. If so, then there are $1 \leq i, j \leq 2$ such that $x_i = y_j = b$, and thus, $u_iv_j = u_i'bbv_j' \notin \alpha(\Gamma^*)$, which is a contradiction. □

Proof of Theorem 2.4. Given a nondeterministic automaton $\mathcal{A}$ over the alphabet $\{a, b, c\}$, we can construct an automaton $\mathcal{A}'$ over $\{a, b\}$ such that $L(\mathcal{A}') = \alpha(L(\mathcal{A}))$ whereas α is the homomorphism from Lemma 6.12, i.e., $\mathsf{sh}(L(\mathcal{A})) = \mathsf{sh}(L(\mathcal{A}'))$. Moreover, we can construct $\mathcal{A}'$ in a way that $\mathcal{A}'$ has at most three times as many states as $\mathcal{A}$. The claim follows from Proposition 6.10. □

6.6 Bibliographic Remarks

Lemma 6.1 is well-known. The notion of a (single) string expression and Lemma 6.2 originate from R.S. Cohen [4]. The definition of the $T_{d,h}(P)$-hierarchy, Lemmas 6.3, 6.4, 6.5, 6.6, and Propositions 6.7 and 6.8 and their proofs originate from the author.

The proof of Theorem 2.3 in Section 6.4 originates from the author. Section 6.5 originates from the author except the notion of the tag property and Theorem 6.11 which are due to K. Hashiguchi and N. Honda [19].

7 On Star Height Substitutions

The aim of this section is to prove Theorem 2.5. During Section 7, let Σ and X be alphabets whereas $\Sigma \cap X = \emptyset$. Let $K \subseteq (\Sigma \cup X)^*$ and $L \subseteq \Sigma^*$ be languages which are recognized by nondeterministic automata $[Q_K, E_K, I_K, F_K]$ resp. $[Q_L, E_L, I_L, F_L]$. Let $\eta : \Sigma^* \to M$ be the syntactic homomorphism and monoid of L. Finally, let $h \geq 0$ be some integer.

7.1 On A, B, C-Substitutions

We develop some preliminaries. Our strategy is to consider at first just nn-substitutions. Then, we generalize our result for nn-substitutions to other kinds of substitutions, in particular, to non-erasing substitutions which are not necessarily non-empty as well as non-empty substitutions which are not necessarily non-erasing, and of course, substitutions which are not necessarily non-empty or non-erasing. To cover all these variants in one unique approach, we introduce the rather technical notion of an A, B, C-substitution.

Let $A, B, C \subseteq X$ be disjoint. We call some substitution $\sigma : X \to \mathcal{P}(\Sigma^*)$ an *A, B, C-substitution*, if we have for every $x \in X$,

1. if $x \in A$, then $\sigma(x) = \emptyset$,
2. if $x \in B$, then $\sigma(x) = \{\varepsilon\}$,
3. if $x \in C$, then $\sigma(x) \supsetneq \{\varepsilon\}$,
4. if $x \in X \setminus (A \cup B \cup C)$, then $\emptyset \neq \sigma(x) \subseteq \Sigma^+$.

For every substitution $\sigma : X \to \mathcal{P}(\Sigma^*)$, there are unique disjoint sets $A, B, C \subseteq X$ such that σ is an A, B, C-substitution. A substitution σ is an nn-substitution iff σ is an $\emptyset, \emptyset, \emptyset$-substitution.

Lemma 7.1. *Let $A, B, C \subseteq X$ be disjoint. If there is an A, B, C-substitution σ such that $\sigma(K) = L$, then there is a recognizable A, B, C-substitution σ' such that $\sigma'(K) = L$ and $\mathsf{sh}(\sigma') \leq |M|$.*

Lemma 7.1 has two benefits. It shows that the problem to decide the existence of a substitution σ such that $\sigma(K) = L$ is equivalent to decide the existence of a recognizable substitution σ such that $\sigma(K) = L$ which is again equivalent to the problem to decide the existence of a substitution σ such that $\sigma(K) = L$ and $\mathsf{sh}(\sigma) \leq |M|$.

Beside this, Lemma 7.1 allows to improve the complexity. Assume $h > |M|$. By Lemma 7.1 there exists a substitution σ such that $\sigma(K) = L$ and $\mathsf{sh}(\sigma) \leq h$ iff there exists a substitution σ such that $\sigma(K) = L$ and $\mathsf{sh}(\sigma) \leq |M|$. Below, this equivalence allows us to optimize our decision algorithm in the case $h \geq |M|$, and in particular, we will achieve an upper complexity bound which is independent on h.

Proof of Lemma 7.1. Let σ be an A, B, C-substitution such that $\sigma(K) = L$. For every $x \in X$, let

$$\sigma'(x) := \begin{cases} \emptyset & \text{if } x \in A, \\ \varepsilon & \text{if } x \in B, \\ \eta^{-1} \circ \eta((\sigma(x)) & \text{if } x \in C, \\ \eta^{-1} \circ \eta((\sigma(x)) \setminus \varepsilon & \text{if } x \in X \setminus (A \cup B \cup C). \end{cases}$$

Every set of the form $\eta^{-1}(P)$ for some $P \subseteq M$ can be recognized by some automaton using M as states. Hence, $\sigma'(x)$ is recognizable and $\mathsf{sh}(\sigma'(x)) \leq |M|$ for every $x \in X$.

For every $x \in A \cup B$, we have $\sigma(x) = \sigma'(x)$. For every $x \in X$, we have $\sigma(x) \subseteq \sigma'(x)$, and hence, $L = \sigma(K) \subseteq \sigma'(K)$. For every $x \in X$, we have $\eta(\sigma'(x)) = \eta(\sigma(x))$. Thus, we have $\eta(\sigma'(K)) = \eta(\sigma(K)) = \eta(L)$, i.e., $\sigma'(K) \subseteq L$.

It is straightforward to verify that σ' is an A, B, C-substitution. □

Again, let $A, B, C \subseteq X$ be disjoint. We define a mapping $\varrho_{A,B,C} : X \to \mathcal{P}(X \cup \varepsilon)$ by

$$\varrho_{A,B,C}(x) := \begin{cases} \emptyset & \text{if } x \in A, \\ \varepsilon & \text{if } x \in B, \\ \varepsilon \cup x & \text{if } x \in C, \\ x & \text{if } x \in X \setminus (A \cup B \cup C). \end{cases}$$

We extend $\varrho_{A,B,C}$ to $\varrho_{A,B,C} : \mathcal{P}((\Sigma \cup X)^*) \to \mathcal{P}((\Sigma \cup X)^*)$ by setting $\varrho_{A,B,C}(a) := a$ for $a \in \Sigma$.

Lemma 7.2. *Let $A, B, C \subseteq X$ be disjoint. The following assertions are equivalent.*

1. *There is an A, B, C-substitution $\sigma : X \to \mathcal{P}(\Sigma^*)$ such that $\sigma(K) = L$ and $\mathsf{sh}(\sigma) \leq h$.*

2. *There is an nn-substitution $\sigma' : (X \setminus A \setminus B) \to \mathcal{P}(\Sigma^*)$ such that $\sigma'(\varrho_{A,B,C}(K)) = L$ and $\mathsf{sh}(\sigma') \leq h$.*

Proof. (2) ⇒ (1) We set $\sigma := \sigma' \circ \varrho_{A,B,C}$. It is straightforward to verify that σ is an A, B, C-substitution and using Lemma 6.1, we easily deduce $\mathsf{sh}(\sigma) = \mathsf{sh}(\sigma') \leq h$.

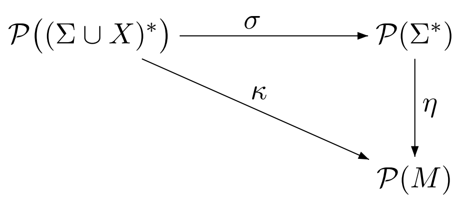
(1) ⇒ (2) For every $x \in X \setminus A \setminus B$, we set $\sigma'(x) := \sigma(x) \setminus \varepsilon$. Clearly, σ is an nn-substitution and we have $\sigma = \sigma' \circ \varrho_{A,B,C}$. Again by Lemma 6.1, we easily deduce $\mathsf{sh}(\sigma) = \mathsf{sh}(\sigma') \leq h$. □

7.2 On Non-empty Non-erasing Substitutions

In this section, we consider nn-substitutions.

Let $\kappa : X \to \mathcal{P}(M)$ be a mapping. By defining $\kappa(a) := \eta(a)$ for every $a \in \Sigma$, κ generalizes to a unique homomorphism $\kappa : \big(\mathcal{P}(\Sigma \cup X)^*, \cup, \cdot\big) \to \big(\mathcal{P}(M), \cup, \cdot\big)$. We call κ a *type* if $\kappa(K) = \eta(L)$ and for every $x \in X$, $\eta^{-1}(\kappa(x))$ contains at least one non-empty word. If κ is a type, then we can show by a counting argument that for every $x \in X$ there is some $w \in \eta^{-1}(\kappa(x))$ such that $|w| \leq |M|$.

Let $\sigma : X \to \mathcal{P}(\Sigma^*)$ be a substitution and set for every $x \in X$, $\kappa(x) := \eta\big(\sigma(x)\big)$. As above, we generalize κ to $\mathcal{P}((\Sigma \cup X)^*)$. For every $a \in \Sigma$, we have $\kappa(a) = \eta(a) = \eta(\sigma(a))$. Hence, we have $\kappa = \eta \circ \sigma$, and in particular, $\eta(\sigma(K)) = \kappa(K)$. If σ is an nn-substitution and $\sigma(K) = L$, then κ is a type.



Lemma 7.3. *The following assertions are equivalent:*

1. *There is an nn-substitution σ' such that $\sigma'(K) = L$ and $\mathsf{sh}(\sigma') \leq h$.*

2. *There is a type κ and a $d \geq |M|$ such that $\sigma(K) = L$ for the nn-substitution σ defined by $\sigma(x) := T_{d,h}\big(\kappa(x)\big)$ for every $x \in X$. Moreover, we have $\mathsf{sh}(\sigma) \leq h$.*

Proof. (2) $\Rightarrow$ (1) We set $\sigma' := \sigma$. Let $x \in X$ be arbitrary. Since $d \geq |M|$ and κ is a type, there is some $w \in \eta^{-1}(\kappa(x))$ such that $1 \leq |w| \leq d$, i.e., $w \in T_{d,0}\big(\kappa(x)\big) \subseteq T_{d,h}\big(\kappa(x)\big) = \sigma'(x)$. By the definition of $T_{d,h}(\kappa(x))$, we have $\varepsilon \notin T_{d,h}(\kappa(x)) = \sigma'(x)$ and $h \geq \mathsf{sh}\big(T_{d,h}(\kappa(x))\big) = \mathsf{sh}(\sigma'(x))$. Hence, σ' is an nn-substitution and $\mathsf{sh}(\sigma') \leq h$.

(1) $\Rightarrow$ (2) For every $x \in X$, we set $\kappa(x) := \eta(\sigma'(x))$. We generalize κ to $\mathcal{P}((\Sigma \cup X)^*)$, and as seen above, we have $\kappa = \eta \circ \sigma'$. Since σ' is an nn-substitution, κ is a type.

Since $\mathsf{sh}(\sigma') \leq h$, there is by Lemma 6.2 for every $x \in X$ a string expression r_x such that $\sigma'(x) = L(r_x)$ and $\mathsf{sh}(r_x) \leq h$. Choose some $d \geq |M|$ such that d is larger than the degrees of r_x for $x \in X$. Since $d \geq |M|$, σ is an nn-substitution. Clearly, $\mathsf{sh}(\sigma) \leq h$.

By Lemma 6.6, we have for every $x \in X$

$$\sigma'(x) \;=\; L(r_x) \;\subseteq\; T_{d,h}\big(\eta(\sigma'(x))\big) \;=\; T_{d,h}\big(\kappa(x)\big) \;=\; \sigma(x),$$

and hence, $\sigma'(K) \subseteq \sigma(K)$, i.e., $L \subseteq \sigma(K)$.

We show $\sigma(K) \subseteq L$. For this, it suffices to show $\eta(\sigma(K)) \subseteq \eta(L)$. Let $x \in X$. By Lemma 6.5, we have $\sigma(x) = T_{d,h}\big(\kappa(x)\big) \subseteq \eta^{-1}(\kappa(x))$, i.e., $\eta(\sigma(x)) \subseteq \kappa(x)$. Hence, $\eta(\sigma(K)) \subseteq \kappa(K)$, and since κ is a type, $\kappa(K) = \eta(L)$. $\square$

Let κ be some type. We show that it is decidable whether there exists a $d \geq |M|$ to satisfy condition (2) in Lemma 7.3. We define a marking $\theta_K : E_K \to \{\curlyvee_0, \ldots, \curlyvee_h, \angle_0, \ldots, \angle_h\}$ for $\mathcal{A}_K$ by setting $\theta(t) = \curlyvee_h$ for every $t \in E_K$.

Let $x \in X$ and $t = (p, x, q) \in E_K$. Let $\mathcal{A}_h\big(\kappa(x)\big)$ be the automaton from Proposition 6.8. We assume that p and q are not states of $\mathcal{A}_h\big(\kappa(x)\big)$. We construct from $\mathcal{A}_h\big(\kappa(x)\big)$ an $(h+1)$-nested distance desert automaton $\mathcal{A}_t = [Q_t, E_t, p, q, \theta_t]$ by renaming the initial and accepting state of $\mathcal{A}_h\big(\kappa(x)\big)$ to p and q, respectively. We assume $Q_K \cap Q_t = \{p, q\}$. Note that if $p = q$, then the initial and accepting state of $\mathcal{A}_h\big(\kappa(x)\big)$ are identified in the construction of $\mathcal{A}_t$.

We assume such an automaton $\mathcal{A}_t$ for every transition $t \in E_K$ which is labeled by a variable. We assume that for distinct transitions $t, t' \in E_K$ the states of $\mathcal{A}_t$ and $\mathcal{A}_{t'}$ are disjoint up to the initial and accepting state which may be common, e.g., if t' starts in the state in which t ends.

We define a nested distance desert automaton $\mathcal{A}_\kappa = [Q_\kappa, E_\kappa, I_\kappa, F_\kappa, \theta_\kappa]$. The states Q_κ are simply the union of the states of $\mathcal{A}_K$ and the states of the automata $\mathcal{A}_t$. We set $I_\kappa := I_K$, $F_\kappa := F_K$. The transitions E_κ are the union of $E_K \cap (Q_K \times \Sigma \times Q_K)$ and the transitions of the automata $\mathcal{A}_t$.

The mapping θ_κ arises in the union of the transitions. However, we should take care that θ_κ is well-defined since the union of the transitions is not necessarily disjoint.

Let (r_1, a, r_2) be a transition in $\mathcal{A}_\kappa$ which belongs to $\mathcal{A}_K$ and to some automaton $\mathcal{A}_t$. We have $\theta_K((r_1, a, r_2)) = \curlyvee_h$, i.e., (r_1, a, r_2) is marked by $\curlyvee_h$ in $\mathcal{A}_K$. Since r_1 is a state in $\mathcal{A}_K$, r_1 is the initial or the accepting state of $\mathcal{A}_t$. If r_1 is the initial state, then $\theta_t((r_1, a, r_2)) = \curlyvee_h$ by the construction of $\mathcal{A}_t$ and (3) in Proposition 6.8. If r_1 is the accepting state of $\mathcal{A}_t$, then (r_1, a, r_2) leaves the accepting state of $\mathcal{A}_t$. By the construction of $\mathcal{A}_t$, and in particular by (1) in Proposition 6.8, the initial and accepting state in $\mathcal{A}_t$ are identical. Hence, r_1 is the initial state of $\mathcal{A}_t$, and as above, $\theta_t((r_1, a, r_2)) = \curlyvee_h$.

Now, let (r_1, a, r_2) be a transition which belongs to two automata $\mathcal{A}_t$ and $\mathcal{A}_{t'}$. By arguing as above, we can conclude that r_1 is the initial state of both $\mathcal{A}_t$ and $\mathcal{A}_{t'}$, and hence, $\theta_t((r_1, a, r_2)) = \theta_{t'}((r_1, a, r_2)) = \curlyvee_h$.

Consequently, θ_κ is well-defined in the union of the transitions.

Proposition 7.4. *Let κ be a type. The following assertions are equivalent.*

1. *There is some $d \geq |M|$ such that we have $\sigma(K) = L$ for the nn-substitution σ defined by $\sigma(x) := T_{d,h}\big(\kappa(x)\big)$ for every $x \in X$.*

2. *We have $L(\mathcal{A}_\kappa) = L$ and $\mathcal{A}_\kappa$ is limited.*

Proof. We state some preliminaries before we prove both directions.

For every $c \geq 1$, let σ_c be the substitution defined by $\sigma_c(x) := T_{c,h}\big(\kappa(x)\big)$ for every $x \in X$.

By Proposition 6.5, we have for every $c \geq 1$, $x \in X$, $\eta(\sigma_c(x)) \subseteq \kappa(x)$. Hence, we have $\eta(\sigma_c(K)) \subseteq \kappa(K) = \eta(L)$, and thus, $\sigma_c(K) \subseteq L$.

We show the following claim $(*)$. To prove $(*)$, we do not assume that (1) or (2) are satisfied.

$(*)$ Let π be an successful path in $\mathcal{A}_\kappa$, denote the label of π by w, and let $c \geq \Delta(\theta_\kappa(\pi)) + 1$. We have $w \in \sigma_c(K)$.

We decompose π. There are some $k \geq 0$ and non-empty paths $\pi_1, \ldots, \pi_k$ in $\mathcal{A}_\kappa$ such that $\pi = \pi_1 \ldots \pi_k$, and for every $1 \leq i \leq k$, the path π_i starts and ends in some state of $\mathcal{A}_K$ but π_i does not contain a state from $\mathcal{A}_K$ inside.

For every $1 \leq i \leq k$, we transform π_i into a path π'_i such that $\pi'_1 \ldots \pi'_k$ is an successful path in $\mathcal{A}_K$, and the label of $\pi'_1 \ldots \pi'_k$ is a word $w' \in K$ such that $w \in \sigma_c(w')$. In particular, we show for every $1 \leq i \leq k$, that σ_c applied to the label of π'_i yields a set which contains the label of π_i.

There are two kinds of paths among $\pi_1, \ldots, \pi_k$: single transitions of $\mathcal{A}_K$ and successful paths of some automaton $\mathcal{A}_t$.

Let $1 \leq i \leq k$, and assume that π_i is a single transition of $\mathcal{A}_K$. We set $\pi'_i := \pi_i$. Since the label of π_i is a letter of Σ, σ_c applied to the label of π'_i yields the label of π_i.

Let $1 \leq i \leq k$, and assume that π_i is not a single transition of $\mathcal{A}$. Hence, there is some transition $t_i = (p_i, x_i, q_i)$ in $\mathcal{A}_K$ such that π_i is an successful path in $\mathcal{A}_{t_i}$. We set $\pi'_i = t_i$. By the definition of the paths $\pi_1, \ldots, \pi_k$, the path π_i does not contain p_i or q_i inside even if p_i and q_i are identified. Hence, π_i is (up to the renamed beginning and ending state) an successful path in $\mathcal{A}_h\big(\kappa(x_i)\big)$. Let w_i be the label of π_i. Since π_i is a factor of π, we have $\Delta(\theta_{t_i}(\pi_i)) + 1 \leq c$. By Proposition 6.8(4), we have $w_i \in T_{c,h}\big(\kappa(x_i)\big) = \sigma_c(x_i)$. Consequently, σ_c applied to the label of π'_i yields a set which contains the label of π_i.

Each path π'_i has the same starting and ending state as π_i. Hence, we can concatenate $\pi'_1, \ldots, \pi'_k$, and $\pi'_1 \ldots \pi'_k$ is an successful path in $\mathcal{A}_K$. The label of $\pi'_1 \ldots \pi'_k$ is some word $w' \in K$ and we have $\sigma_c(w') = w$ which completes the proof of $(*)$.

(2) $\Rightarrow$ (1) Let $d \geq |M|$ such that for every $w \in L(\mathcal{A}_\kappa) = L$, we have $\Delta(w) + 1 \leq d$. By $(*)$, we have for every $w \in L$, $w \in \sigma_d(K)$, i.e., $L \subseteq \sigma_d(K)$. Above, we have already shown $\sigma_d(K) \subseteq L$. Since $d \geq |M|$ and κ is a type, σ_d is an nn-substitution.

(1) $\Rightarrow$ (2) Let $w \in L(\mathcal{A}_\kappa)$ be arbitrary. By $(*)$, there is some $c \geq 1$ such that $w \in \sigma_c(K)$. Above, we have shown $\sigma_c(K) \subseteq L$ for every $c \geq 1$. Hence, $L(\mathcal{A}_\kappa) \subseteq L$.

It remains to show that $L \subseteq L(\mathcal{A}_\kappa)$ and that $\mathcal{A}_\kappa$ is limited. Let $w \in L$ be arbitrary. It suffices to construct a successful path π in $\mathcal{A}_\kappa$ such that π is labeled with w and $\Delta(\theta_\kappa(\pi)) + 1 \leq d$.

Since $w \in L = \sigma_d(K)$, there is some $w' \in K$ such that $w \in \sigma_d(w')$. Denote $w' = b_1 \ldots b_{|w'|}$, where $b_1, \ldots, b_{|w'|} \in (\Sigma \cup X)$. For $1 \leq i \leq |w'|$, let $w_i \in \sigma_d(b_i)$ such that $w_1 \ldots w_{|w'|} = w$. Let π' be an successful path in $\mathcal{A}_K$ and denote $\pi' = t_1 \ldots t_{|w'|}$ whereas $t_1, \ldots, t_{|w'|}$ are transitions.

Let $1 \le i \le |w'|$. If $b_i \in \Sigma$, then set $\pi_i = t_i$. Assume $b_i \in X$, and denote $t_i = (p_{i-1}, b_i, p_i)$. We have $w_i \in \sigma_d(b_i) = T_{d,h}(\kappa(b_i))$. By the construction of $\mathcal{A}_{t_i}$ and Proposition 6.8(4), there is an successful path π_i in $\mathcal{A}_{t_i}$ which is labeled with w_i and $\Delta(\theta_{t_i}(\pi_i)) + 1 \le d$.

The concatenation $\pi := \pi_1 \dots \pi_{|w'|}$ is an successful path in $\mathcal{A}'$ which is labeled with w. It remains to show $\Delta(\theta_\kappa(\pi)) + 1 \le d$. There are no transitions which are marked by $\angle_{h+1}$. Let $0 \le g \le h$, and let $\tilde{\pi}$ be a factor of π in which we cannot obtain g-coins. By Proposition 6.8(3), and since every transition of $\mathcal{A}_K$ is labeled with Υ_h, $\tilde{\pi}$ is a factor of some path π_i, and hence, $\Delta(\theta_\kappa(\tilde{\pi})) + 1 \le d$. Consequently, $\Delta(\theta_\kappa(\pi)) + 1 \le d$. □

Proposition 7.5. *Given $h \ge 0$, languages $K \subseteq (\Sigma \cup X)^*$ and $L \subseteq \Sigma^*$ which are recognized by nondeterministic automata $[Q_K, E_K, I_K, F_K]$ and $[Q_L, E_L, I_L, F_L]$ it is decidable whether there exists an nn-substitution $\sigma : X \to \mathcal{P}(\Sigma^*)$ satisfying $\sigma(K) = L$ and $\mathsf{sh}(\sigma) \le h$. There is an algorithm to decide the existence of σ whose space complexity is polynomial in $|E_K| \cdot 2^{2^{\mathcal{O}(|Q_L|^2)}}$.*

Proof. We sketch a deterministic algorithm.

At first, the algorithm constructs the syntactic monoid M of L. It has at most $2^{|Q_L|^2}$ elements.

Then, the algorithm proceeds the following computations for every mapping $\kappa : X \to \mathcal{P}(M)$ such that for every $x \in X$, $\kappa(x) \ne \emptyset$.

It checks whether $\kappa(K) = \eta(L)$. If so, the algorithm checks whether for every $x \in X$, there is some non-empty word in $\eta^{-1}(\kappa(x))$. For this, the algorithm computes the set $\eta(\Sigma^+) = \langle \eta(\Sigma) \rangle$, and checks whether for every $x \in X$, the intersection $\kappa(x) \cap \eta(\Sigma^+)$ is not empty.

If κ is not a type, then the algorithm continues with the next mapping κ.

If κ is a type, then the algorithm constructs $\mathcal{A}_\kappa$, and it decides whether $L = L(\mathcal{A}_\kappa)$ and $\mathcal{A}_\kappa$ is limited. If so, the algorithm answers "yes". Otherwise, the algorithm continues with the next mapping κ.

If the algorithm answers "yes", then (2) in Proposition 7.4 is satisfied. By (1) in Proposition 7.4, the desired nn-substitution σ exists.

Conversely, assume that there is some nn-substitution σ such that $\sigma(K) = L$ and $\mathsf{sh}(\sigma) \le h$. Hence, (1) and (2) in Lemma 7.3 are true. We consider the behavior of the algorithm for the type κ in Lemma 7.3(2). By Lemma 7.3(2), Proposition 7.4(1) is true for κ, and by (2) of Proposition 7.4, the algorithm answers "yes".

To analyze the space complexity, we consider two optimizations. The algorithm can ignore variables which do not occur in $\mathcal{A}_K$. Hence, we can assume $|X| \le |E_K|$.

If $h > |M|$, then we know from Lemma 7.1 that there exists an nn-substitution σ such that $\sigma(K) = L$ and $\mathsf{sh}(\sigma) \le h$ iff there exists an nn-substitution σ' such that $\sigma'(K) = L$ and $\mathsf{sh}(\sigma') \le |M|$. Hence, if $h > |M|$, then the algorithm can carry out the involved construction of $\mathcal{A}_\kappa$ for $|M|$ instead of h. Consequently, we can assume $h \le |M|$.

We can assume that every state in $\mathcal{A}_K$ occurs in at least one transition of $\mathcal{A}_K$ except one state in $I_K \cap F_K$ to accept the empty word. Hence, we can assume $|Q_K| \le 2|E_K| + 1$.

To store a mapping $\kappa : X \to \mathcal{P}(M)$, the algorithm requires $|X| \cdot |M|$, i.e., $|E_K| \cdot 2^{|Q_L|^2}$ space. To compute $\kappa(K)$ to check whether $\kappa(K) = \eta(L)$, the algorithm can iterate over a $Q_K \times |M|$-array, i.e., it requires $|Q_K| \cdot 2^{|Q_L|^2}$ space. To compute $\langle \eta(\Sigma) \rangle$, it requires just $|M| \le 2^{|Q_L|^2}$ space. To sum up, the algorithm requires $|E_K| \cdot 2^{\mathcal{O}(|Q_L|^2)}$ space to store κ and to check whether κ is a type.

By Proposition 6.8(2), each automaton $\mathcal{A}_t$ has at most $2^{|M|h}\big(|M|+2\big)$ states. We insert at most $|E_K|$ automata $\mathcal{A}_t$. Since $h \le |M| \le 2^{|Q_L|^2}$, $\mathcal{A}_\kappa$ has at most

$$|E_K| \cdot \big(2^{|M|^2}\big) \cdot \big(|M| + 1\big) \;\le\; |E_K| \cdot \Big(2^{2^{(2|Q_L|^2)}}\Big) \cdot \big(2^{|Q_L|^2} + 1\big)$$

states. We can then test in polynomial space whether $L = L(\mathcal{A}_\kappa)$ and by Theorem 2.2 it is decidable in polynomial space whether $\mathcal{A}_\kappa$ is limited. □

7.3 On Arbitrary Recognizable Substitutions

Now, we consider arbitrary recognizable substitutions, i.e., recognizable substitutions which are not necessarily non-erasing or non-empty. At first, we generalize Proposition 7.5 to A, B, C-substitutions.

Proposition 7.6. *Given mutually disjoint sets $A, B, C \subseteq X$, $h \geq 0$, languages $K \subseteq (\Sigma \cup X)^*$ and $L \subseteq \Sigma^*$ which are recognized by nondeterministic automata $[Q_K, E_K, I_K, F_K]$ and $[Q_L, E_L, I_L, F_L]$ it is decidable whether there exists an A, B, C-substitution $\sigma : X \to \mathcal{P}(\Sigma^*)$ satisfying $\sigma(K) = L$ and $\mathsf{sh}(\sigma) \leq h$. There is an algorithm to decide the existence of σ whose space complexity is polynomial in $|E_K| \cdot 2^{2^{\mathcal{O}(|Q_L|^2)}}$.*

Proof. By Lemma 7.2, there is an A, B, C-substitution $\sigma : X \to \mathcal{P}(\Sigma^*)$ such that $\sigma(K) = L$ and $\mathsf{sh}(\sigma) \leq h$ iff there is an nn-substitution $\sigma' : (X \setminus A \setminus B) \to \mathcal{P}(\Sigma^*)$ such that $\sigma'(\varrho_{A,B,C}(K)) = L$ and $\mathsf{sh}(\sigma') \leq h$.

Hence, an algorithm constructs an automaton $\mathcal{A}'_K$ which recognizes $\varrho_{A,B,C}(K)$ and applies Proposition 7.5. The construction of $\mathcal{A}'_K$ can be carried out as follows:

1. We construct a set of transitions $E_1 := E_K \cap \big(Q \times (X \setminus A) \times Q\big)$.

2. We construct a set of transitions E_2. A transition $(p, b, q) \in E_1$ (whereas $b \in (\Sigma \cup X)$) belongs to E_2 iff there are a transition $(p', b, q') \in E_1$ and a path π_1 (resp. π_2) over E_1 from p to p' (resp. q' to q) such that both π_1 and π_2 are labeled with words in $(B \cup C)^*$. Note that π_1 and π_2 may be empty, and hence, $E_1 \subseteq E_2$.

3. We construct a set of transitions $E_3 := E_2 \cap \big(Q \times (X \setminus B) \times Q\big)$.

If $\varepsilon \in \varrho_{A,B,C}(K) \setminus K$, then we introduce a new state q and set $\mathcal{A}'_K = [Q_K \uplus q,\ E_3,\ I_K \uplus q,\ F_K \uplus q]$. Otherwise, we set $\mathcal{A}'_K = [Q_K, E_3, I_K, F_K]$. □

Proof of Theorem 2.5. By Proposition 7.5, we have (2) for non-empty, non-erasing substitutions. To show (2) for non-empty substitutions, we simply decide by Proposition 7.6 for every disjoint sets $B, C \subseteq X$ the existence of a $\emptyset, B, C$-substitution σ which satisfies the desired properties. For non-erasing substitutions, we decide for every $A \subseteq X$ the existence of an $A, \emptyset, \emptyset$-substitution, and for (1) we consider for every mutually disjoint sets A, B, C the existence of an A, B, C-substitution satisfying the desired properties. □

7.4 Some Remarks on the Complexity

Let us consider the constructions from Section 7 for $h = 0$, i.e., let us consider finite substitutions. Then, we can construct the automaton $\mathcal{A}_\kappa$ in Section 7.2 in a more efficient way, because by Proposition 7.6, each automaton $\mathcal{A}_0(\kappa(x))$ has at most $|M| + 2$ states. Hence, $\mathcal{A}_\kappa$ has at most $|E_K|\big(|M| + 2\big)$ states which reduces the space complexity in Propositions 7.5 and 7.6 to $|E_K| \cdot 2^{\mathcal{O}(|Q_L|^2)}$. Thus, we can decide the existence of σ in EXPSPACE as S. Bala in [1, 2].

Now, let us consider the constructions from Section 7 for recognizable substitutions of arbitrary star height, i.e., let us consider the *recognizable substitution problem.* We can construct exactly the same automaton $\mathcal{A}_\kappa$ as in the case $h = 0$. Then, we can show as an adaptation of Lemma 7.3 that there is a recognizable substitution σ' satisfying $\sigma'(K) = L$ iff there is a type κ such that $\sigma_\kappa(K) = L$ for the substitution defined by $\sigma_\kappa(x) := \eta^{-1}(\kappa(x))$. As in Proposition 7.2, we can show $\sigma_\kappa(K) = L$ iff $L = L(\mathcal{A})$. As for $h = 0$, we can reduce the space complexity in Propositions 7.5 and 7.6 to $|E_K| \cdot 2^{\mathcal{O}(|Q_L|^2)}$ as long as we are just interested in a recognizable substitution rather than a substitution of a certain star height. Thus, we can decide the existence of σ in EXPSPACE as S. Bala in [1, 2]. In particular, limitedness of $\mathcal{A}_\kappa$ does not play a role if we are just interested in the recognizable substitution problem.

7.5 Bibliographic Remarks

The finite substitution problem (i.e. substitutions of star height 0) was already raised by J.-E. Pin around 1992 [49]. The technical notion of an A, B, C-substitution and the constructions in Section 7.1 originate from the author. For the particular case $h = 0$, the underlying ideas in construction of the automaton $\mathcal{A}_\kappa$ in Section 7.2, Lemma 7.3, and Propositions 7.4 and 7.5 were already shown independently by S. Bala and the author [1, 2, 22, 25]. Proposition 7.6 and the proof of Theorem 2.5 are due to the author.

8 Conclusions and Challenges

In the authors opinion, there are two challenges concerning desert automata and the star height problem.

The first challenge is to determine the exact complexity of the star height problem. In particular, it is not clear whether its reduction to limitedness of nested distance desert automata can be achieved in a more efficient way.

The other challenge is an extension of our concepts to achieve decidability results for other hierarchies of classes of recognizable languages, e.g., the Straubing-Thérien hierarchy, the dot-depth hierarchy, and the famous extended star height hierarchy [47]. It is not clear whether or how our principle of using nested distance desert automata to examine languages $T_{d,h}(P)$ can be generalized to decide language hierarchies which allow complement and intersection. Maybe, one needs to develop more involved automata concepts than nested distance desert automata.

Beside these two challenges, there are several other things to investigate.

As pointed out in Section 2.4, the decidability of the equivalence of two desert automata (1-nested distance desert automata in which transitions are marked by $\curlyvee_0$ and $\angle_0$) is an open question.

Another open question is to give a sharp bound on the range of the mappings of limited nested distance desert automata depending on the number of states. For limited n-state distance automata, the sharpest known upper bound on the range is $2^{3n^3+n\lg n+n-1}$ [35], but the worst known examples are limited by 2^n-2 [33, 54].

The limitedness problem for distance automata was originally motivated by the star height problem, but it turned out to be useful in other areas, e.g., [9, 26, 40]. At this point, there are two applications of desert and nested distance desert automata: the decidability of the finite substitution problem [22, 25] and a new proof for the decidability of the star height problem in the present paper. One should look for other applications and establish connections between nested distance desert automata and other concepts in theoretical computer science.

Teil II
Deutsche Kurzfassung (Short Version in German)

Dieser Teil enthält die deutsche Kurzfassung der Habilitationsschrift gemäß § 6 Abs. 2 der Habilitationsordnung der Fakultät für Mathematik und Informatik der Universität Leipzig vom 9. Juli 1998.

Distanzdesertautomaten und Sternhöhensubstitutionen

Zusammenfassung

Wir führen verschachtelte Disanzdesertautomaten als eine gemeinsame Verallgemeinerung von Distanzautomaten und Desertautomaten ein. Wir zeigen, dass die Beschränktheit von verschachtelten Disanzdesertautomaten PSPACE-vollständig ist.

Als Anwendung zeigen wir, dass es zu einer gegebenen Zahl h und einem nichtdeterministischen Automaten mit n Zuständen mit Speicherplatz in $2^{2^{\mathcal{O}(n^2)}}$ entscheidbar ist, ob die Sternhöhe der akzeptierten Sprache kleiner gleich h ist. Weiterhin zeigen wir die Entscheidbarkeit einiger Probleme von Substitutionen erkennbarer Sprachen.

1 Einführung

Das *Sternhöhenproblem* wurde 1963 von L.C. Eggan formuliert [7]: Ist die Sternhöhe erkennbarer Sprachen berechenbar? Er bezog sich dabei auf "klassische" Sternhöhe bezüglich Vereinigung, Verkettung und Iteration ohne Komplement und Schnittmengenbildung. Das Sternhöhenproblem wurde lange Zeit als das schwierigste Problem in der Theorie der erkennbaren Sprachen betrachtet. Es dauerte 25 Jahre bis K. Hashiguchi die Berechenbarkeit der Sternhöhe zeigte [14]. Dieses Ergebnis wird als eines der bedeutendsten Ergebnisse in der Formalen Sprachtheorie angesehen. Der Artikel [14] ist jedoch sehr schwer zu verstehen. J.-É. Pin bemerkte: "Hashiguchis Lösung für beliebige Sternhöhe beruht auf einer komplizierten Induktion, so dass man dem Beweis nur sehr schwer folgen kann"[1] [48]. Hashiguchis Ansatz erstreckt sich über [11, 12, 13, 14] und I. Simon kommentierte, dass man für eine umfassende Darstellung des gesamten Beweises eine "mehr als einhundert Seiten lange, sehr schwierige kombinatorische Argumentation"[1] führen muss [51]. D. Perrin schrieb: "Der Beweis ist sehr schwierig zu verstehen und es muss noch viel getan werden, um ihn im Stil eines Lehrbuchs darzustellen"[1] [43].

Aus K. Hashiguchis Ansatz erhält man einen Algorithmus von nicht elementarer Komplexität, und es bis heute nicht gelungen, eine obere Komplexitätsschranke anzugeben (s. [36, Anhang B]).

Zur Lösung des Sternhöhenproblems führte K. Hashiguchi 1982 Distanzautomaten ein [11, 12]. Ein *Distanzautomat* ist ein nichtdeterministischer, endlicher Automat, bei dem einige Transitionen markiert sind. Das Gewicht eines Pfades ist die Anzahl der markierten Transitionen in dem Pfad. Das Gewicht eines Wortes w ist das Minimum der Gewichte aller erfolgreichen Pfade für w. K. Hashiguchi zeigte 1982, dass die Beschränktheit von Distanzautomaten entscheidbar ist, d.h. es ist entscheidbar, ob es eine obere Schranke für die Gewichte aller akzeptierten Worte gibt.

Distanzautomaten erlangten eine große Bedeutung in der Theoretischen Informatik, bsw. in der Forschung zum Sternproblem für Spurmonoide [26, 40], aber auch für Anwendungen wie Spracherkennung [41], Datenbanken [9] und Bildkompression [5, 21], so dass Distanzautomaten und verwandte Konzepte intensiv untersucht wurden [15, 17, 27, 31, 35, 50, 51, 53, 54, 55].

Desertautomaten wurden 2004 unabhängig von S. Bala und dem Autor eingeführt [1, 2, 22, 25]. Ein *Desertautomat* ist ein nichtdeterministischer, endlicher Automat, bei dem einige Transitionen markiert sind. Das Gewicht eines Pfades π ist die Länge eines längsten Teilpfades, in dem keine markierten Transitionen vorkommen. Das Gewicht eines Wortes w ist das Minimum der Gewichte aller erfolgreichen Pfade für w. S. Bala und der Autor zeigten 2004 unabhängig voneinander, dass die Beschränktheit von Desertautomaten entscheidbar ist [1, 2, 22, 25]. Dadurch kann man die Entscheidbarkeit des *finite-substitution Problems* zeigen, welches zuvor mehr als 10 Jahre offen war.

In der Habilitationsschrift führen wir eine gemeinsame Verallgemeinerung und Weiterentwicklung von Distanz- und Desertautomaten ein, die sog. *verschachtelten Distanzdesertautomaten*. Durch Weiterentwicklung der Ansätze aus [15, 22, 23, 25, 29, 30, 34, 50, 51, 53] zeigen wir, dass die Beschränktheit von verschachtelten Distanzdesertautomaten PSPACE-vollständig ist. Für die Entscheidbarkeit in PSPACE beantworten wir eine Frage aus H. Leungs Dissertation [29] von 1987.

Als Anwendung verschachtelter Distanzdesertautomaten zeigen wir einen neuen Beweis und die erste obere Komplexitätsschranke für das Sternhöhenproblem: Zu einer gegebenen Zahl h und einem nichtdeterministischen Automaten mit n Zuständen ist es mit Speicherplatz in $2^{2^{\mathcal{O}(n^2)}}$ entscheidbar, ob die Sternhöhe der akzepierten Sprache kleiner gleich h ist. Weiterhin zeigen wir die Entscheidbarkeit einiger Probleme von Substitutionen erkennbarer Sprachen.

[1] Übersetzungen des Autors.

2 Überblick

2.1 Notationen, Automaten und Sprachen

Die Numerierung von Sätzen, Theoremen, ... in dieser Kurzfassung ist aus der Habilitationsschrift übernommen und daher nicht fortlaufend.

Es sei $\mathbb{N} = \{0, 1, \dots\}$. Für eine Menge M bezeichen $\mathcal{P}(M)$ bzw. $\mathcal{P}_{ne}(M)$ die Potenzmenge von M bzw. die Menge aller nichtleeren Teilmengen von M.

Eine *Halbgruppe* $(S, \cdot)$ ist eine Menge mit einer binären assoziativen Operation "$\cdot$". Ein Element 1 in einer Halbgruppe S heißt *neutrales Element*, wenn für alle $p \in S$ $1a = a1 = a$ gilt. Falls S ein neutrales Element besitzt, dann bezeichnen wir S als *Monoid*.

Für zwei Elemente $p, q \in S$ definiert man $p \leq_{\mathscr{J}} q$, falls $p \in SqS \cup qS \cup Sq \cup q$ gilt. Falls $p \leq_{\mathscr{J}} q$ und $q \leq_{\mathscr{J}} p$ gilt, dann sind p und q *$\mathscr{J}$-äquivalent* und man schreibt $p =_{\mathscr{J}} q$. Die Äquivalenzklassen von $=_{\mathscr{J}}$ heißen *$\mathscr{J}$-Klassen*.

Eine Teilmenge von S, die unter $\cdot$ abgeschlossen ist, heißt *Unterhalbgruppe* von S. Für Teilmengen $T \subseteq S$ bezeichnet $\langle T \rangle$ die kleinste Unterhalbgruppe von S, die T enthält.

Es sei Σ eine endliche Menge von Symbolen. Wir bezeichnen das freie Monoid bzw. die freie Halbgruppe über Σ mit Σ^* bzw. Σ^+. Wir bezeichnen das leere Wort mit ε. Für $w \in \Sigma^*$ bezeichnet $|w|$ die Länge von w. Für $L \subseteq \Sigma^*$, seien $L^* := L^0 \cup L^1 \cup \cdots = \cup_{i \in \mathbb{N}} L^i$ und $L^+ := L^1 \cup L^2 \cup \cdots = \cup_{i \geq 1} L^i$.

Ein (nichtdeterministischer) *Automat* ist ein Tupel $\mathcal{A} = [Q, E, I, F]$ wobei Q eine endliche Menge von *Zuständen*, $E \subseteq Q \times \Sigma \times Q$ eine Menge von Transitionen und $I \subseteq Q$, $F \subseteq Q$ die Initial- bzw. akzeptierenden Zustände sind. Es sei $k \geq 1$. Ein Pfad π in $\mathcal{A}$ der Länge k ist eine Folge $(q_0, a_1, q_1)\,(q_1, a_2, q_2) \dots (q_{k-1}, a_k, q_k)$ von Transitionen in E. Wir notieren $|\pi| := k$. Das Wort $a_1 \dots a_k$ ist die *Beschriftung* von π. Falls $q_0 \in I$ und $q_k \in F$, dann bezeichnen wir π als *erfolgreich*.

Für $P, R \subseteq Q$ und $w \in \Sigma^*$ bezeichnen wir mit $P \overset{w}{\rightsquigarrow} R$ die Menge aller Pfade in $\mathcal{A}$, die in einem Zustand in P beginnen, in einem Zustand in R enden und mit w beschriftet sind.

Es seien π_1 und π_2 Pfade in $\mathcal{A}$. Die Verkettung $\pi_1\pi_2$ ist definiert, wenn π_2 in dem Zustand beginnt, in dem π_1 endet. Ein Pfad π' ist ein *Faktor* eines Pfades π, wenn es Pfade π_1, π_2 gibt, so dass die Verkettung $\pi_1\pi'\pi_2$ definiert ist und $\pi = \pi_1\pi'\pi_2$ ist. Jeder Pfad ist ein Faktor von sich selbst.

Die von $\mathcal{A}$ erkannte Sprache besteht aus allen Beschriftungen erfolgreicher Pfade in $\mathcal{A}$ und wird mit $L(\mathcal{A})$ notiert. Eine Sprache $L \subseteq \Sigma^*$ ist *erkennbar*, wenn L von einem Automaten erkannt wird. Zum Einstieg in die Theorie der erkennbaren Sprachen wird der Leser an [3, 8, 56] verwiesen.

Ein Monoid M *erkennt* eine Sprache $L \subseteq \Sigma^*$, wenn es einen Homomorphismus $\eta : \Sigma^* \to M$ gibt, so dass $L = \eta^{-1}(\eta(L))$ gilt. Die Sprache L ist genau dann erkennbar, wenn L von einem endlichen Monoid erkannt wird. Das kleinste Monoid, welches L erkennt, wird als das *syntaktische Monoid* von L bezeichnet. Zu einem Automaten $\mathcal{A} = [Q, E, I, F]$ kann man das syntaktische Monoid M von $L(\mathcal{A})$ effektiv konstruieren und es gilt $|M| \leq 2^{|Q|^2}$.

Der Begriff eines $\sharp$-Ausdrucks wurde 1990 von K. Hashiguchi eingeführt [15]. Jeder Buchstabe $a \in \Sigma$ ist ein $\sharp$-Ausdruck. Für $\sharp$-Ausdrücke r und s sind rs und $r^\sharp$ ebenfalls $\sharp$-Ausdrücke.

Jeder $\sharp$-Ausdruck r definiert für $k \geq 1$ ein Wort $r(k)$. Für $r \in \Sigma$ gilt $r(k) := r$. Weiterhin gilt $rs(k) := r(k) \cdot s(k)$. Für $r^\sharp$ definieren wir $r^\sharp(k)$ als die k-fache Kopie von $r(k)$, d.h. $r^\sharp(k) := \big(r(k)\big)^k$.

Die $\sharp$-Höhe von $\sharp$-Ausdrücken ist induktiv definiert. Buchstaben haben die $\sharp$-Höhe 0. Die $\sharp$-Höhe von rs ist die maximale $\sharp$-Höhe von r bzw. s, und die $\sharp$-Höhe von $r^\sharp$ ist die $\sharp$-Höhe von r plus 1.

Es sei X ein endliches, zu Σ disjunktes Alphabet. Abbildungen $\sigma : X \to \mathcal{P}(\Sigma^*)$ bezeichnen wir als *Substitutionen*. Durch $\sigma(a) := a$ für $a \in \Sigma$ setzt sich jede Substitution zu einem eindeutigen

Homomorphismus $\sigma : (\Sigma \cup X)^* \to \mathcal{P}(\Sigma^*)$ fort. Eine Substitution σ heißt *nichtlöschend* (bzw. *nichtleer* bzw. *endlich* bzw. *erkennbar*), wenn für alle $x \in X$ $\sigma(x) \subseteq \Sigma^+$ gilt (bzw. $\sigma(x) \neq \emptyset$ gilt bzw. $\sigma(x)$ endlich ist bzw. $\sigma(x)$ erkennbar ist).

2.2 Verschachtelte Distanzdesertautomaten

Es seien $h \in \mathbb{N}$ und $V := \{\angle_0, \curlyvee_0, \angle_1, \curlyvee_1, \ldots, \curlyvee_{h-1}, \angle_h\}$. Wir definieren eine Abbildung $\Delta : V^* \to \mathbb{N}$. Zunächst erläutern wir die Grundidee. Für jedes $0 \leq g \leq h$ gibt es Münzen der Farbe g, die wir als g-Münzen bezeichnen. Wir haben einen Beutel um Münzen zu transportieren. Der Beutel hat genau $h+1$ Fächer, die wie die Münzen gefärbt sind. Für jede Farbe $0 \leq g \leq h$ können wir g-Münzen im Fach g aufbewahren, aber wir können g-Münzen nicht in einem anderen Fach aufbewahren. In einem Beutel der Größe $d \in \mathbb{N}$ können wir jeweils d Münzen jeder Farbe aufbewahren. Somit können wir zwar insgesamt $(h+1)d$ Münzen transportieren, aber wir können höchstens d Münzen von ein und derselben Farbe transportieren.

Nun planen wir, entlang einem Wort[2] $\pi \in V^*$ zu laufen. Wir haben einen Beutel der Größe d. Der Beutel ist am Anfang vollständig gefüllt. Wenn wir an einem Buchstaben $\angle_g$ vorbeikommen, müssen wir eine g-Münze bezahlen, aber wir können beliebig viele Münzen der Farben $0, \ldots, g-1$ erhalten. Wir spechen $\angle_g$ als "péage g" aus. Falls wir keine g-Münze haben, dann können wir $\angle_g$ nicht passieren. Wenn wir an einem Buchstaben $\curlyvee_g$ vorbeikommen, dann brauchen wir nichts zu bezahlen, aber wir können beliebig viele Münzen der Farben $0, \ldots, g$ erhalten. Wir spechen $\curlyvee_g$ als "water g" aus.

Es hängt von der Größe des Beutels (und von π) ab, ob wir das gesamte Wort π passieren können. Es sei $\Delta(\pi)$ die kleinste Zahl d, so dass ein Beutel der Größe d ausreichend ist, um π passieren zu können.

Wir definieren Δ formal. Es sei $0 \leq g \leq h$. Wir betrachten alle Faktoren von π, in denen wir keine g-Münzen erhalten können, d.h. wir betrachten Faktoren π' von π mit $\pi' \in \{\angle_0, \curlyvee_0, \ldots, \angle_g\}^*$ und zählen die Anzahl der Buchstaben $\angle_g$ in π'. Diese ist die Anzahl der g-Münzen, die entlang π' benötigt werden.

Für $0 \leq g \leq h$ und $\pi' \in V^*$ sei $|\pi'|_g$ die Anzahl der Buchstaben $\angle_g$ in π'. Nun sei

1. $\Delta_g(\pi) := \max_{\substack{\pi' \in \{\angle_0, \curlyvee_0, \ldots, \angle_g\}^* \\ \pi' \text{ ist ein Faktor von } \pi}} |\pi'|_g$ und
2. $\Delta(\pi) := \max_{0 \leq g \leq h} \Delta_g(\pi)$.

Ein *h-fach verschachtelter Distanzdesertautomat* ist ein Tupel $\mathcal{A} = [Q, E, I, F, \theta]$, wobei $[Q, E, I, F]$ ein endlicher Automat und θ eine Abbildung $\theta : E \to V$ ist. Wir erweitern θ zu einem Homomorphismus $\theta : E^* \to V^*$. Für Pfade π ist $\theta(\pi)$ die *Markierung* von π. Für $w \in \Sigma^*$ sei

$$\Delta_{\mathcal{A}}(w) := \min_{\pi \in I \overset{w}{\rightsquigarrow} F} \Delta(\theta(\pi)).$$

Falls es ein $d \in \mathbb{N}$ gibt, so dass $\Delta_{\mathcal{A}}(w) \leq d$ für alle $w \in L(\mathcal{A})$ gilt, dann bezeichnen wir $\mathcal{A}$ als *beschränkt*, anderenfalls als *unbeschränkt*.

K. Hashiguchis Distanzautomaten [11] sind 1-fach verschachtelte Distanzdesertautomaten mit der Einschränkung $\theta : E \to \{\curlyvee_0, \angle_1\}$. Die von S. Bala und dem Autor eingeführten Desertautomaten [1, 2, 22, 25] sind genau die 1-fach verschachtelten Distanzdesertautomaten mit der Einschränkung $\theta : E \to \{\curlyvee_0, \angle_0\}$.

[2] Wir verwenden π sowohl für Worte in V^* als auch für Pfade in Automaten.

2.3 Hauptergebnisse

Ein wichtiges Hauptergebnis der Habilitationsschrift ist die folgende zweifache Charakterisierung unbeschränkter verschachtelter Distanzdesertautomaten:

Theorem 2.1. *Es seien $h \in \mathbb{N}$ und $\mathcal{A} = [Q, E, I, F, \theta]$ ein h-fach verschachtelter Distanzdesertautomat. Die folgenden Aussagen sind äquivalent:*

1. *$\mathcal{A}$ ist unbeschränkt.*
2. *Es sei $T := \Psi(\Sigma)$. Es gibt eine Matrix $a \in \langle T \rangle^\sharp$, so dass $I \cdot a \cdot F = \omega$ gilt.*
3. *Es gibt einen $\sharp$-Ausdruck mit einer $\sharp$-Höhe von höchstens $(h+1)|Q|$, so dass für alle $k \geq 1$ $r(k) \in L(\mathcal{A})$ gilt und für wachsende $k \geq 1$ das Gewicht $\Delta_\mathcal{A}\big(r(k)\big)$ unbeschränkt ist.*

Die algebraischen Konzepte in (2) und der Beweis (2)⇒(3) werden in Kapitel 3 behandelt. Die Implikation (3)⇒(1) gilt offensichtlich. Der Beweis vom (1)⇒(2) führt auf ein sog. Burnside-Problem und wird in dieser Kurzfassung nicht behandelt.

Theorem 2.1 verallgemeinert Charakterisierungen unbeschränkter Distanz- bzw. Desertautomaten von K. Hashiguchi, H. Leung, I. Simon und dem Autor [15, 22, 25, 29, 30, 34, 50, 51, 53]. Die Beschränkung der $\sharp$-Höhe in (3) ist jedoch auch für Distanzautomaten ein neues Ergebnis.

Aus Theorem 2.1 erhalten wir:

Theorem 2.2. *Für $h \geq 1$ ist die Beschränktheit von h-fach verschachtelten Distanzdesertautomaten PSPACE-vollständig.*

Theorem 2.2 verallgemeinert Resultate von H. Leung und V. Podolskiy [35] bzw. S. Bala und dem Autor [1, 2, 22, 25] für Distanz- bzw. Desertautomaten. Der Beweis beruht jedoch nicht auf einer Verallgemeinerung dieser Speziallfälle, sondern auf einer neuen Idee, wobei u.a. eine Frage aus H. Leungs Dissertation [29] von 1987 beantwortet wird (vgl. Korollar 5.6(2) in Kapitel 3.4).

Als Anwendung von Theorem 2.2 erhalten wir neue Ergebnisse zum Sternhöhenproblem erkennbarer Sprachen. Die *Sternhöhe* eines rationalen Ausdrucks notieren wir mit $\mathsf{sh}(r)$. Für alle $w \in \Sigma^*$ gilt $\mathsf{sh}(w) := 0$ sowie $\mathsf{sh}(\emptyset) := 0$. Für Ausdrücke r und s definieren wir $\mathsf{sh}(rs) = \mathsf{sh}(r \cup s) := \mathsf{max}\big\{\mathsf{sh}(r), \mathsf{sh}(s)\big\}$ und $\mathsf{sh}(r^*) := \mathsf{sh}(r) + 1$.

Für alle $k \in \mathbb{N}$ sei nun $\mathcal{L}_k := \big\{\, L(r) \,\big|\, \mathsf{sh}(r) \leq k \,\big\}$. Damit ist $\mathcal{L}_0$ die Klasse aller endlichen Sprachen. Die *Sternhöhe* einer erkennbaren Sprache L ist das kleinste $k \in \mathbb{N}$ mit $L \in \mathcal{L}_k$ und wird mit $\mathsf{sh}(L)$ notiert. Bereits 1963 zeigte L.C. Eggan die Striktheit der Inklusion $\mathcal{L}_k \subsetneq \mathcal{L}_{k+1}$ für alle $k \in \mathbb{N}$, doch er benötigte dazu ein Alphabet mit $2^{k+1} - 1$ Buchstaben [7]. In dem gleichen Artikel erwähnte er das *Sternhöhenproblem*:

1. Ist die Inklusion $\mathcal{L}_k \subseteq \mathcal{L}_{k+1}$ strikt für alle $k \in \mathbb{N}$ für $\Sigma = \{a, b\}$?
2. Ist die Sternhöhe erkennbarer Sprachen berechenbar?

Bereits 1966 zeigten F. Dejean and M. Schützenberger die Striktheit der Inklusion für $\Sigma = \{a, b\}$ und beantworteten damit die erste Frage [6]. 1982 zeigte K. Hashiguchi, dass es entscheidbar ist, ob eine erkennbare Sprache die Sternhöhe 1 hat [12, 13]. 1988 konnte er schließlich zeigen, dass die Sternhöhe erkennbarer Sprachen berechenbar ist [14]. Die Probleme von K. Hashiguchis Ansatz wurden bereits in der Einleitung erwähnt.

Der Autor zeigte mit Hilfe von verschachtelten Distanzdesertautomaten die erste obere Komplexitätsschranke für das Sternhöhenproblem [24]. Hier zeigen wir das folgende Theorem:

Theorem 2.3. *Es seien $h \geq 1$ und L die Sprache eines nichtdeterministischen Automaten $\mathcal{A}$ mit n Zuständen. Es ist mit Speicherplatz in $2^{2^{\mathcal{O}(n^2)}}$ entscheidbar, ob die Sternhöhe von L kleiner gleich h ist.*

Die Komplexität in Theorem 2.3 hängt nicht von h ab, da die Sternhöhe von L stets kleiner gleich n ist und somit im Fall $h \geq n$ keine Berechnungen notwendig sind. Wir zeigen in Kapitel 4.4:

Theorem 2.4. *Es sei $h \geq 1$. Das Problem, ob die Sternhöhe der Sprache eines nichtdeterministischen Automaten über einem zweibuchstabigen Alphabet kleiner gleich h ist, ist PSPACE-hart.*

Man kann die Komplexität in Theorem 2.3 auf $2^{2^{\mathcal{O}(n)}}$ verbessern [24]. Mit der hier entwickelten Konstruktion lassen sich jedoch über das Sternhöhenproblem hinaus einige Probleme erkennbarer Substitutionen lösen. Die *Sternhöhe* einer erkennbaren Substitution $\sigma : X \to \mathcal{P}(\Sigma^*)$ ist die maximale Sternhöhe der Sprachen $\sigma(x)$ für alle $x \in X$.

Theorem 2.5.

1. *Zu gegebenen $h \geq 0$, Sprachen $K \subseteq (\Sigma \cup X)^*$ und $L \subseteq \Sigma^*$, die von nicht-deterministischen Automaten $[Q_K, E_K, I_K, F_K]$ und $[Q_L, E_L, I_L, F_L]$ erkannt werden, ist es entscheidbar, ob es eine erkennbare Substitution $\sigma : X \to \mathcal{P}(\Sigma^*)$ mit $\sigma(K) = L$ und $\mathsf{sh}(\sigma) \leq h$ gibt.*

   Dieses Problem ist in einer Speicherplatzkomplexität von $|E_K| \cdot 2^{2^{\mathcal{O}(|Q_L|^2)}}$ entscheidbar.

2. *Wie in (1) kann man in der gleichen Komplexität entscheiden, ob es eine nichtlöschende (bzw. nichtleere bzw. nichtlöschende nichleere) Substitution $\sigma : X \to \mathcal{P}(\Sigma^*)$ mit $\sigma(K) = L$ und $\mathsf{sh}(\sigma) \leq h$ gibt.*

Der Autor entwickelte Theorem 2.5 mit dem Ziel, mehrere bekannte Resultate in einen einzigen Ansatz zusammenzufassen. Insbesondere ist der Spezialfall $K = \{x\}$ in Theorem 2.5(1) eine Lösung des Sternhöhenproblems.

Da endliche Substitutionen genau die erkennbaren Substitutionen mit einer Sternhöhe 0 sind, enthält Theorem 2.5 für $h = 0$ die Entscheidbarkeit des sog. *finite substitution problems*, welches bereits Anfang der 90-iger Jahre von J.-E. Pin betrachtet wurde [49]. S. Bala zeigte 2004, dass dieses Problem EXPSPACE-vollständig ist [1, 2]. Unabhängig davon zeigte der Autor die Entscheidbarkeit einer etwas schwächeren Variante [22, 25].

Wie in Thereom 2.5 kann man auch die Existenz einer erkennbaren Substitution beliebiger Sternhöhe oder die Existenz einer beliebigen Substitution σ mit $\sigma(K) = L$ untersuchen. S. Bala zeigte, dass diese Probleme EXPSPACE-vollständig sind [1, 2]. Es gibt jedoch genau dann eine Substitution σ mit $\sigma(K) = L$, wenn es eine erkennbare Substitution σ' mit $\sigma'(K) = L$ und $\mathsf{sh}(\sigma') \leq 2^{|Q_L|^2}$ gibt. Daher erhalten wir S. Balas Resultate aus Theorem 2.5 für $h = 2^{|Q_L|^2}$, allerdings mit einer deutlich schlechteren Komplexitätsschranke. In der Habilitationsschrift wird gezeigt, wie man die Beweiskonstruktion für Theorem 2.5 in diesen beiden Spezialfällen optimieren kann, um die Entscheidbarkeit in EXPSPACE zu zeigen.

3 Das Beschränktheitsproblem

Im gesamten Kapitel 3 seien $h \in \mathbb{N}$ und $\mathcal{A} = [Q, E, I, F, \theta]$ ein h-fach verschachtelter Distanzdesertautomat. Wir setzen $n := |Q|$ und nehmen $Q = \{1, \ldots, n\}$ an.

3.1 Der Semiring $\mathcal{V}$

Wir entwickeln einen Semiring $\mathcal{V}$, um verschachtelte Distanzdesertautomaten zu beschreiben.

Es seien $h \in \mathbb{N}$ und $\mathcal{V} = V \cup \{\omega, \infty\}$. Wir betrachten die folgende Ordnungsrelation

$$\angle_0 \sqsubseteq \Upsilon_0 \sqsubseteq \angle_1 \sqsubseteq \Upsilon_1 \sqsubseteq \ldots \sqsubseteq \Upsilon_{h-1} \sqsubseteq \angle_h \sqsubseteq \omega \sqsubseteq \infty$$

auf $\mathcal{V}$. Wir definieren eine Multiplikation $\cdot$ auf $\mathcal{V}$ als das Maximum bezüglich $\sqsubseteq$. Es sei $\psi : V^+ \to \mathcal{V}$ der kanonische Homomorphismus.

Es sei $\pi \in V^+$. Wir sagen, dass wir π *in einem Zyklus durchlaufen können*, wenn es ein $d \in \mathbb{N}$ gibt, so dass für alle $k \in \mathbb{N}$ $\Delta(\pi^k) \leq d$ gilt. Eine grundlegende Eigenschaft von ψ ist es, dass wir π genau dann in einem Zyklus durchlaufen können, wenn $\psi(\pi) \in \{\Upsilon_0, \ldots, \Upsilon_{h-1}\}$ gilt. Nehmen wir an, es gilt $\psi(\pi) = \angle_g$ für ein $0 \leq i \leq h$. Dann enthält π den Buchstaben $\angle_i$, d.h. wir müssen entlang π g-Münzen bezahlen. Aufgrund der Definition von $\sqsubseteq$ und wegen $\psi(\pi) = \angle_g$ können wir jedoch entlang π keine g-Münzen bekommen. Somit gilt für alle $k \in \mathbb{N}$ $\Delta(\pi^k) \geq k$.

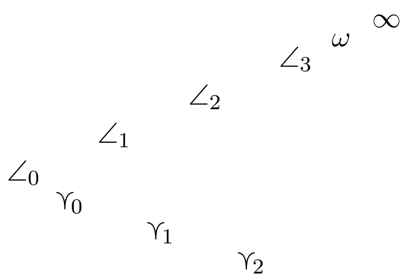
Wenn aber $\psi(\pi) = \Upsilon_g$ für ein $0 \leq g < h$ gilt, dann erhalten wir entlang von π alle die Münzen, die wir entlang von π bezahlen müssen. Man kann zeigen, dass für alle $k \in \mathbb{N}$ $\Delta(\pi^k) \leq 2\Delta(\pi)$ gilt.

Nun betrachten wir die folgende Relation $\leq$ auf $\mathcal{V}$:

$$\Upsilon_{h-1} \leq \Upsilon_{h-2} \leq \ldots \leq \Upsilon_0 \leq \angle_0 \leq \ldots \leq \angle_h \leq \omega \leq \infty.$$

Die Relation $\leq$ beschreibt, welche Transitionen wir bevorzugen. Haben wir die Wahl zwischen zwei Transitionen, die mit Υ_g und Υ_{g-1} (für ein $0 < g < h$) beschriftet sind, dann wählen wir die mit Υ_i markierte Transition, weil wir dort $0, \ldots, g$-Münzen bekommen, an der anderen Transition jedoch nur $0, \ldots, (g-1)$-Münzen. Haben wir die Wahl zwischen zwei Transitionen, die mit $\angle_g$ und $\angle_{g+1}$ (für ein $0 \leq g < h$) markiert sind, dann wählen wir die Transition, die mit $\angle_g$ markiert ist, weil wir $g+1$-Münzen wertvoller als g-Münzen ansehen. Wir definieren auf $\mathcal{V}$ die Operation min als Minimum bezüglich $\leq$.

Das folgende Diagramm zeigt die Relationen $\sqsubseteq$ und $\leq$ für $h = 3$, wobei $\sqsubseteq$ als "links von" und $\leq$ als "unterhalb von" dargestellt ist.



Für beliebige $x, y, z \in \mathcal{V}$ mit $x \leq y$ gilt $xz \leq yz$. Somit gilt das Distributivgesetz und $(\mathcal{V}, \mathsf{min}, \cdot)$ ist ein Semiring. Wir bezeichnen den Semiring der $n \times n$-Matrizen über $\mathcal{V}$ mit $\mathcal{V}_{n \times n}$.

Wir definieren einen Homomorphismus $\Psi : \Sigma^+ \to \mathcal{V}_{n \times n}$ durch

$$\Psi(w)[i,j] := \mathsf{min}\left\{\psi(\theta(\pi)) \,\middle|\, \pi \in i \overset{w}{\rightsquigarrow} j\right\}$$

für alle $w \in \Sigma^+$ und i, j.

Der Untersemiring von $\mathcal{V}$ über der Menge $\mathcal{R} = \{\Upsilon_0, \angle_1, \omega, \infty\}$ wurde von I. Simon und H. Leung zur Entscheidbarkeit der Beschränktheit von Distanzautomaten benutzt [29, 30, 34, 51, 53]. Ebenso wurde der Untersemiring von $\mathcal{V}$ über der Menge $\mathcal{D} = \{\angle_0, \Upsilon_0, \omega, \infty\}$ vom Autor zur Entscheidbarkeit der Beschränktheit von Desertautomaten eingesetzt [22, 25].

3.2 Stabilisierung

Wie in Kapitel 3.1 gesehen, kann man das Verhalten eines verschachtelten Distanzdesertautomaten mit Matrizen über $\mathcal{V}$ beschreiben. Wir führen nun eine Operation $\sharp$ ein, um das Verhalten von verschachtelten Distanzdesertautomaten auf Folgen von Worten zu beschreiben.

Für alle $z \in \mathcal{V}$ sei

$$z^\sharp := \begin{cases} z & \text{if } z \in \{\Upsilon_0, \ldots, \Upsilon_{h-1}\} \\ \omega & \text{if } z \in \{\angle_0, \ldots, \angle_h, \omega\} \\ \infty & \text{if } z = \infty. \end{cases}$$

Für alle $z \in \mathcal{V}$ gilt $z \leq z^\sharp$ und $zz^\sharp = z^\sharp z = z^\sharp$.

Wir definieren eine Abbildung $\sharp : \mathsf{E}(\mathcal{V}_{n\times n}) \to \mathsf{E}(\mathcal{V}_{n\times n})$, die wir als *Stabilisierung* bezeichnen. Für alle $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ und i, j sei

$$e^\sharp[i,j] \;:=\; \mathsf{min}_{1\leq l\leq n} \big(e[i,l] \cdot (e[l,l])^\sharp \cdot e[l,j]\big).$$

Diese Definition ist eine gemeinsame Verallgemeinerung der Stabilisierung von I. Simon und H. Leung für idempotente Matrizen in $\mathcal{R}_{n\times n}$ [29, 30, 34, 51, 53] und der vom Autor entwickelten Stabilisierung idempotenter Matrizen in $\mathcal{D}_{n\times n}$ [22, 25].

Für Teilmengen $T \subseteq \mathcal{V}_{n\times n}$ definieren wir $\langle T\rangle^\sharp$ als die kleinste Teilmenge von $\mathcal{V}_{n\times n}$, die T enthält und sowohl unter Multiplikation von Matrizen als auch unter Stabilisierung $\sharp$ idempotenter Matrizen abgeschlossen ist. Da $\langle T\rangle^\sharp$ endlich und berechenbar ist, kann man (2) in Theorem 2.1 entscheiden.

Es seien $w \in \Sigma^+$ und $e := \Psi(w) \in \mathsf{E}(\mathcal{V}_{n\times n})$. Es gilt dann für alle $k \geq 1$, $\Psi(w^k) = e$. Intuitiv beschreibt $e^\sharp$ das Verhalten von $\mathcal{A}$ auf der Folge w, w^2, w^3, … Es seien i, j beliebig.

1. Falls $e^\sharp[i,j] = \infty$, dann gilt auch $e[i,j] = \infty$. In diesem Fall kann $\mathcal{A}$ keines der Worte w, w^2, w^3, … von i nach j lesen.

2. Falls $e^\sharp[i,j] \in \{\Upsilon_0, \angle_1, \ldots, \Upsilon_{h-1}, \angle_h\}$, dann gibt es ein $d \in \mathbb{N}$ und für alle $k \geq 3$ einen Pfad $\pi \in i \overset{w^k}{\rightsquigarrow} j$ mit $\Delta(\theta(\pi)) \leq d$ und $\psi(\theta(\pi)) = e^\sharp[i,j]$.

   Man kann π wie folgt konstruieren: Es sei $c \in \mathbb{N}$, so dass es zu jedem i', j' mit $e[i',j'] \notin \{\omega, \infty\}$ einen Pfad $\pi \in i' \overset{w}{\rightsquigarrow} j'$ mit $\Delta(\theta(\pi)) \leq c$ und $\psi(\theta(\pi)) = e[i',j']$ gibt.

   Es sei l, so dass $e^\sharp[i,j] = e[i,l] \cdot (e[l,l])^\sharp \cdot e[l,j]$ gilt. Weil $e^\sharp[i,j] \notin \{\omega, \infty\}$ ist, gilt $(e[l,l])^\sharp \notin \{\omega, \infty\}$ und somit $(e[l,l])^\sharp = e[l,l] \in \{\Upsilon_0, \ldots, \Upsilon_{h-1}\}$.

   Nun seien $\pi_1 \in i \overset{w}{\rightsquigarrow} l$, $\pi_2 \in l \overset{w}{\rightsquigarrow} l$ und $\pi_3 \in l \overset{w}{\rightsquigarrow} j$, so dass $\Delta(\theta(\pi_1)) \leq c$, $\psi(\theta(\pi_1)) = e[i,l]$ und analoge Bedingungen für π_2, π_3 gelten. Man kann dann $\pi := \pi_1(\pi_2)^{k-1}\pi_3$ setzen und erhält $\Delta(\theta(\pi)) \leq 4c$, d.h. die obige Ausage gilt für $d = 4c$.

3. Der Fall $e^\sharp[i,j] = \angle_0$ ist nicht möglich, da für alle l $\Upsilon_0 \sqsubseteq (e[l,l])^\sharp$ gilt.

Im Allgemeinen gilt $e[i,j] \leq e^\sharp[i,j]$. Betrachten wir nun den interessanteren Fall $e[i,j] < e^\sharp[i,j]$. Wir wollen weiterhin annehmen, dass $e[i,j] \notin \{\omega, \infty\}$. Es gilt dann $e^\sharp[i,j] \neq \infty$. Es sei $k \geq 3$.

Falls $e^\sharp[i,j] \neq \omega$, dann gibt es wie in (2) einen Pfad $\pi \in i \overset{w^k}{\rightsquigarrow} j$ mit $\Delta(\theta(\pi)) \leq d$ und $\psi(\theta(\pi)) = e^\sharp[i,j]$. Wegen $\Psi(w^k) = e$ gibt es jedoch auch einen Pfad $\pi' \in i \overset{w^k}{\rightsquigarrow} j$ mit $\psi(\theta(\pi')) = e[i,j] < e^\sharp[i,j]$. Wir zerlegen π' in $\pi' = \pi_1 \dots \pi_p$, so dass jeder der Pfade $\pi_0, \dots, \pi_p$ mit einem Vielfachen von w beschriftet ist und jeder der Pfade $\pi_1, \dots, \pi_{p-1}$ ein Zylkus ist. Wir können dabei $p \geq \frac{k-1}{n}$ annehmen. Wenn für ein $0 < i < p$ $\psi(\theta(\pi_i)) \in \{\Upsilon_0, \dots, \Upsilon_{h-1}\}$ gilt, dann kann man $e^\sharp[i,j] \leq \Psi(\theta(\pi'))$ zeigen, was der Wahl von π' widerspräche. Somit gilt für alle $0 < i < p$ $\psi(\theta(\pi_i)) \in \{\angle_0, \dots, \angle_h\}$. Damit kann man zeigen, dass $\Delta(\theta(\pi')) \geq \sqrt[h+1]{\frac{k-1}{n} - 1} - 1$ gilt, d.h. wir haben für das Gewicht von π' eine *untere* Schranke, die von k abhängt.

Diese untere Schranke gilt für jeden Pfad $\pi' \in i \overset{w^k}{\rightsquigarrow} j$ mit $\psi(\theta(\pi')) < e^\sharp[i,j]$. Im Spezialfall $e^\sharp[i,j] = \omega$ gilt $\psi(\theta(\pi')) < e^\sharp[i,j]$ jedoch für jeden Pfad $\pi' \in i \overset{w^k}{\rightsquigarrow} j$, so dass $\mathcal{A}$ die Worte w^k von i nach j zwar lesen kann, aber nur mit einem Gewicht welches mit k unbeschränkt ansteigt.

3.3 Über $\sharp$-Ausdrücke

Wir definieren den Typ einiger $\sharp$-Ausdrücke. Jeder Buchstabe $a \in \Sigma$ ist ein $\sharp$-Ausdruck vom *Typ* $\tau(a) = \Psi(a)$. Sind r und s zwei $\sharp$-Ausdrücke, deren Typen $\tau(r)$ und $\tau(s)$ definiert sind, dann ist rs ein $\sharp$-Ausdruck vom *Typ* $\tau(rs) = \tau(r)\tau(s)$. Ist r ein $\sharp$-Ausdruck, $\tau(r)$ definiert und gilt $\tau(r) \in \mathsf{E}(\mathcal{V}_{n\times n})$, dann ist $r^\sharp$ ein $\sharp$-Ausdruck vom *Typ* $\tau(r^\sharp) = \big(\tau(r)\big)^\sharp$. Falls der Typ eines $\sharp$-Ausdrucks r definiert ist, dann heißt r ein *typisierter $\sharp$-Ausdruck*.

Für jeden typisierten $\sharp$-Ausdruck r gilt offenbar $\tau(r) \in \langle T\rangle^\sharp$. Umgekehrt kann man zu jeder Matrix $p \in \langle T\rangle^\sharp$ einen typisierten $\sharp$-Ausdruck r mit $\tau(r) = p$ konstruieren.

Die Beobachtungen in Kapitel 3.2 lassen sich auf typisierte $\sharp$-Ausdrücke verallgemeinern.

Satz 4.3. *Es sei r ein typisierter $\sharp$-Ausdruck mit $I \cdot \tau(r) \cdot F = \omega$. Für alle $k \geq 1$ gilt $r(k) \in L(\mathcal{A})$. Das Gewicht $\Delta_{\mathcal{A}}\big(r(k)\big)$ wächst unbeschränkt für ansteigende Zahlen k.*

Beweisidee. Man zeigt folgende Aussage induktiv über den Aufbau von $\sharp$-Ausdrücken:

Es sei r ein typisierter $\sharp$-Ausdruck.
Für jede Schranke $d \geq 0$ gibt es ein $K \geq 1$, so dass für alle $k \geq K$ gilt:
Für alle i, j und alle Pfade $\pi \in i \overset{r(k)}{\rightsquigarrow} j$ mit $\psi(\theta(\pi)) < \tau(r)[i,j]$ gilt $\Delta(\theta(\pi)) \geq d$. □

In Satz 4.3 kann man den Beweis von (2)⇒(3) in Theorem 2.1 erkennen. Es ist aber noch nicht klar, wie man einen geeigneten $\sharp$-Ausdruck r mit einer $\sharp$-Höhe von höchstens $(h+1)n$ konstruiert.

3.4 Zur Komplexität

In diesem Kapitel skizzieren wir algebraische Argumente, mit denen man die Beschränktheit von verschachtelten Distanzdesertautomaten in PSPACE entscheiden kann (Theorem 2.2). Weiterhin können wir damit im Beweis von (2)⇒(3) in Theorem 2.1 die $\sharp$-Höhe von r beschränken.

Es sei $T \subseteq \mathcal{V}_{n\times n}$ und $T_0 := \langle T\rangle$. Für alle $p \in \mathbb{N}$ definieren wir:

$$T_{p+1} := \Big\langle\, T_p \,\cup\, \big\{e^\sharp \,\big|\, e \in \mathsf{E}(T_p)\big\} \,\Big\rangle$$

Wir bezeichnen $T_0 \subseteq T_1 \subseteq T_2 \dots$ als die *Stabilisierungshierarchie von T*. Es gilt $\langle T\rangle^\sharp = \bigcup_{p\geq 0} T_p$ und $\langle T\rangle^\sharp = T_{|\mathcal{V}_{n\times n}|}$, wobei $|\mathcal{V}_{n\times n}| = (2h+3)^{n^2}$.

Eine zentrale Frage bei der Komplexität des Beschränktheitsproblems von verschachtelten Distanzdesertautomaten ist: In welcher Ebene kollabiert die Stabilisierungshierarchie? Für Distanzautomaten wurde diese Frage bereits 1987 in der Dissertation von H. Leung aufgeworfen [34].

H. Leung vermutete, dass für $\mathcal{R} = \{\curlyvee_0, \angle_1, \omega, \infty\}$ und $T \subseteq \mathcal{R}_{n\times n}$ $\langle T\rangle^\sharp = T_{n^2}$ gilt [29, S. 38]. 1988 vermutete man, dass es ein Polynom $B : \mathbb{N} \to \mathbb{N}$ gibt, so dass $\langle T\rangle^\sharp = T_{B(n)}$ für alle $T \subseteq \mathcal{R}_{n\times n}$ gilt [51, S. 112]. 1991 hat H. Leung die Existenz eines derartigen Polynoms B jedoch wieder als ein offenes Problem dargestellt [32, S. 522]. Die große Bedeutung dieses Problems resultiert daraus, dass man mit Hilfe eines derartigen Polynoms B die Beschränktheit von Distanzautomaten in PSPACE entscheiden kann. Man benutzt dabei B als eine Art Abbruchbedingung in einem Algorithmus, der die Hülle $\langle T\rangle^\sharp$ durchsucht, um (2) in Theorem 2.1 zu entscheiden.

Bereits 1991 bewies H. Leung, dass das Polynom $B(n) := n - 2$ nicht ausreichend ist [32].

H. Leung und V. Podolskiy zeigten 2004 unter Benutzung von Ideen aus [16, 17, 18], dass die Beschränktheit von Distanzautomaten in PSPACE entscheidbar ist [35]. Das obige Polynom B spielt in diesem Ansatz keine Rolle und man kann keine Rückschlüsse auf dessen Existenz ziehen.

Ein Grundansatz zur Beantwortung von H. Leungs Frage bestünde darin, eine Abbildung $\mathsf{grd} : \mathcal{R}_{n\times n} \to \{0, \ldots, n^2\}$ einzuführen, die folgende Eigenschaften erfüllt:

1. Für alle $a, b \in \mathcal{R}_{n\times n}$ gilt $\mathsf{grd}(ab) \leq \min\{\mathsf{grd}(a), \mathsf{grd}(b)\}$.

2. Für alle $e \in \mathsf{E}(\mathcal{R}_{n\times n})$ mit $e \neq e^\sharp$ gilt $\mathsf{grd}(e^\sharp) < \mathsf{grd}(e)$.

Der Haken dabei ist, dass wegen (1) für alle Matrizen $a, b \in \mathcal{R}_{n\times n}$ mit $a \leq_{\mathscr{J}} b$ stets $\mathsf{grd}(a) \leq \mathsf{grd}(b)$ gelten muss. Somit muss die Abbildung grd auf $\mathscr{J}$-Klassen von $\mathcal{R}_{n\times n}$ invariant sein. Da es jedoch $\mathscr{J}$-äquivalente Matrizen gibt, die völlig verschieden zu sein scheinen, ist es schwierig grd so zu definieren, dass (1) und (2) erfüllt sind. Insbesondere scheitern an dieser Beobachtung naheliegende Ansätze, wie z.B. grd durch Zählen von ω-Einträgen oder durch gewichtete Summen über die Einträge zu definieren. Die Lösung dieses Problems ist erstaunlich einfach.

Es seien $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ und $0 \leq g \leq h$. Wir definieren eine Relation $\sim_{e,g}$ auf $\{1, \ldots, n\}$ durch

$$i \sim_{e,g} j \qquad :\Longleftrightarrow \qquad e[i,j] \leq \angle_g \text{ und } e[j,i] \leq \angle_g$$

für alle i, j. Weil e idempotent ist, ist $\sim_{e,g}$ transitiv und symmetrisch. Auf der Menge

$$Z_{e,g} := \{i \mid \text{es gibt ein } j, \text{ so dass } i \sim_{e,g} j\}$$

ist $\sim_{e,g}$ eine Äquivalenzrelation. In der Habilitationschrift werden die folgenden Aussagen bewiesen:

1. Es seien $0 \leq g \leq h$ sowie $e, f \in \mathsf{E}(\mathcal{V}_{n\times n})$ mit $f \leq_{\mathscr{J}} e$. Die Relation $\sim_{f,g}$ hat höchstens so viele Äquivalenzklassen wie $\sim_{e,g}$ (Lemma 5.1).

2. Es seien $0 \leq g \leq h$ und $e \in \mathsf{E}(\mathcal{V}_{n\times n})$. Jede Äquivalenzklasse von $\sim_{e^\sharp,g}$ ist auch eine Äquivalenzklasse von $\sim_{e,g}$ (Lemma 5.2).

3. Es sei $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ mit $e \neq e^\sharp$. Es gibt ein $0 \leq g \leq h$, so dass $\sim_{e^\sharp,g}$ echt weniger Äquivlenzklassen als $\sim_{e,g}$ hat (Lemma 5.3).

Somit kann man für Matrizen $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ $\mathsf{grd}(e)$ definieren, indem man über alle $0 \leq g \leq h$ die Anzahl der Äquivalenzklassen von $\sim_{e,g}$ summiert. Da $\sim_{e,g}$ höchstens n Äquivalenzklassen hat, gilt für alle $e \in \mathsf{E}(\mathcal{V}_{n\times n})$ $0 \leq \mathsf{grd}(e) \leq (h+1)n$. Dann zeigt man mittels (1)(2)(3), dass für alle $p \geq 1$ und alle idempotenten Matrizen $e \in T_p \setminus T_{p-1}$ gilt $\mathsf{grd}(e) \leq (h+1)n - p$. Somit kann die Stabilisierung idempotenter Matrizen nicht aus $T_{(h+1)n}$ herausführen, und wir erhalten:

Korollar 5.6.

1. *Es sei* $h \geq 1$ *und* $T \subseteq \mathcal{V}_{n\times n}$. *Es gilt* $T_{(h+1)n} = \langle T\rangle^\sharp$.
2. *Für alle* $T \subseteq \mathcal{R}_{n\times n}$ *gilt* $T_n = \langle T\rangle^\sharp$.

Insbesondere beantwortet Korollar 5.6(2) H. LEUNGs Frage, da $T_n \subseteq T_{n^2} \subseteq \langle T\rangle^\sharp$ und damit $T_{n^2} = \langle T\rangle^\sharp$ gilt. Nach Korollar 5.6(2) kann man für das oben erwähnte Polynom $B(n) := n$ setzen.

In Kapitel 5.4 der Habilitationsschrift wird aufbauend auf Korollar 5.6(1) ein Algorithmus entwickelt, der $\langle T\rangle^\sharp$ durchsucht, jedoch nur polynomiell viele Matrizen gleichzeitig im Speicher hält. Damit kann man (2) in Theorem 2.1 und die Beschränktheit in PSPACE entscheiden.

Wir können nun den Beweis von (2)⇒(3) in Theorem 2.1 abschließen.

Beweis von (2)⇒(3) in Theorem 2.1. Zu jedem $p \in \mathbb{N}$, $a \in T_p$ kann man induktiv über p einen typisierten $\sharp$-Ausdruck r mit $\tau(r) = a$ konstruieren, so dass die $\sharp$-Höhe von r höchstens p beträgt. Es sei $a \in \langle T\rangle^\sharp$ mit $I \cdot a \cdot F = \omega$. Nach Korollar 5.6(1) gilt $a \in T_{(h+1)|Q|}$. Aus Satz 4.3 für einen typisierten $\sharp$-Ausdruck r mit $\tau(r) = a$ und einer $\sharp$-Höhe von höchstens $(h+1)|Q|$ folgt (3). □

4 Das Sternhöhenproblem

In diesem Kapitel skizzieren wir die Beweise der Theoreme 2.3 und 2.4.

Es sei $h \in \mathbb{N}$. Weiterhin seien M ein endliches Monoid und $\eta : \Sigma^* \to M$ ein surjektiver Homomorphismus. Wir erweitern η zu einem eindeutigen Homomorphismus $\eta : \mathcal{P}(\Sigma^*) \to \mathcal{P}(M)$.

Wir wollen die Sternhöhe der Sprachen $\eta^{-1}(P)$ für Teilmengen $P \subseteq M$ untersuchen.

4.1 Stringausdrücke

Zunächst zeigen wir:

Lemma 6.1. *Für jede erkennbare Sprache* $L \subseteq \Sigma^*$ *gilt* $\mathsf{sh}(L) = \mathsf{sh}(L \setminus \varepsilon)$.

Wir übernehmen den Begriff der Stringausdrücke von R.S. COHEN [4]. Wir definieren die Begriffe Stringausdruck, einzelner Stringausdruck, und Grad eines Stringausdrucks in einer simultanen Induktion. Jedes Wort $w \in \Sigma^*$ ist ein *einzelner Stringausdruck* der Sternhöhe $\mathsf{sh}(w) = 0$ und vom *Grad* $\mathsf{dg}(w) := 0$. Für $n \geq 1$ und einzelne Stringausdrücke $r_1, \ldots, r_n$, ist $r := r_1 \cup \cdots \cup r_n$ ein *Stringausdruck* mit $\mathsf{sh}(r) = \mathsf{max}\{\mathsf{sh}(r_i) \mid 1 \leq i \leq n\}$ und vom *Grad* $\mathsf{dg}(r) := \mathsf{max}\{\mathsf{dg}(r_i) \mid 1 \leq i \leq n\}$. Die leere Menge $\emptyset$ ist ein *Stringausdruck* der Sternhöhe $\mathsf{sh}(\emptyset) = 0$ und vom *Grad* $\mathsf{dg}(\emptyset) := 0$.

Es seien $n \geq 2$, $a_1, \ldots, a_n \in \Sigma$ und $s_1, \ldots, s_{n-1}$ Stringausdrücke. Wir bezeichnen den Ausdruck $s := a_1 s_1^* a_2 s_2^* \ldots s_{n-1}^* a_n$ als *einzelnen Stringausdruck* der Sternhöhe $\mathsf{sh}(s) = 1 + \mathsf{max}\{\mathsf{sh}(s_i) \mid 1 \leq i < n\}$ vom *Grad* $\mathsf{dg}(s) := \mathsf{max}\big(\{n\} \cup \{\mathsf{dg}(s_i) \mid 1 \leq i < n\}\big)$.

Es bezeichne $L(r)$ die Sprache des Stringausdrucks r.

Das folgende Lemma stammt von R.S. COHEN [4].

Lemma 6.2. *Jede erkennbare Sprache* $L \subseteq \Sigma^*$ *wird von einem Stringausdruck* r *mit* $\mathsf{sh}(r) = \mathsf{sh}(L)$ *erzeugt.*

Um zu ermitteln, ob die Sternhöhe einer Sprache $\eta^{-1}(P)$ kleiner gleich h ist, betrachten wir die nach Lemma 6.1 und 6.2 äquivalente Frage, ob $\eta^{-1}(P) \setminus \varepsilon$ durch einen Stringausdruck von einer Sternhöhe höchstens h erzeugbar ist.

4.2 Die $T_{d,h}(P)$-Hierarchie

Für Teilmengen $P, R \subseteq M$ definieren wir $R^{-1}P := \{p \in M \mid Rp \subseteq P\}$. Für beliebige $P, R \subseteq M$, gilt somit $R(R^{-1}P) \subseteq P$. Für alle $P \subseteq M$ ist $P^{-1}P$ ein Untermonoid von M, welches in der Halbgruppentheorie als der *Rechtsstabilisator von P* bezeichnet wird.

Es seien $d \geq 1$ und $P \subseteq M$. Wir definieren $T_{1,0}(P) := \{a \in \Sigma \mid \eta(a) \in P\}$ und für $d > 1$

$$T_{d,0}(P) := \bigcup_{\substack{\text{Für alle } 1 \leq c \leq d \text{ und} \\ P_0,\ldots,P_c \subseteq M, \text{ wobei } P_0 = \{1\}, P_c \subseteq P.}} T_{1,0}\big(P_0^{-1}P_1\big)\, T_{1,0}\big(P_1^{-1}P_2\big) \cdots T_{1,0}\big(P_{c-1}^{-1}P_c\big).$$

Lemma 6.3. *Für alle $d \geq 1$ und $P \subseteq M$ gilt $T_{d,0}(P) = \{w \mid \eta(w) \in P,\ 1 \leq |w| \leq d\}$.*

Aus Lemma 6.3 folgt insbesondere $\bigcup_{d \geq 1} T_{d,0}(P) = \eta^{-1}(P) \setminus \varepsilon$.

Es sei $h \in \mathbb{N}$. Induktiv sei für alle $P \subseteq M$ die Menge $T_{d,h}(P)$ bereits definiert. Wir definieren

$$T_{d,\,h+1}(P) := \bigcup_{\substack{\text{Für alle } 1 \leq c \leq d \text{ und} \\ P_0,\ldots,P_c \subseteq M, \text{ wobei } P_0 = \{1\}, P_c \subseteq P.}} T_{1,0}(P_0^{-1}P_1)\Big(T_{d,h}(P_1^{-1}P_1)\Big)^* T_{1,0}(P_1^{-1}P_2)\Big(T_{d,h}(P_2^{-1}P_2)\Big)^* \cdots T_{1,0}(P_c^{-1}P_c).$$

Es seien $d \geq 1$, $h \in \mathbb{N}$ und $P \subseteq M$ beliebig. Aus der Definition folgt $T_{d,h}(P) \subseteq T_{d,h}(P')$ für alle $P \subseteq P' \subseteq M$. Durch Induktion über h kann man leicht zeigen, dass für alle $d \leq d'$ $T_{d,h}(P) \subseteq T_{d',h}(P)$ sowie $T_{d,h}(P) \subseteq T_{d,\,h+1}(P)$ gilt. Somit gilt für alle $1 \leq d \leq d'$, $0 \leq h \leq h'$ und $P \subseteq P' \subseteq M$ die Inklusion $T_{d,h}(P) \subseteq T_{d',h'}(P')$.

Wenn wir von der $T_{d,h}(P)$*-Hierarchie* sprechen, dann sehen wir h und P als fest an und betrachten die eindimensionale Hierarchie bezüglich d.

Induktiv über h kann man zu jedem $d \geq 1$, $h \in \mathbb{N}$ und $P \subseteq M$ einen Stringausdruck r konstruieren, so dass $L(r) = T_{d,h}(P)$ sowie $\mathsf{dg}(r) \leq d$ und $\mathsf{sh}(r) \leq h$ gilt. Somit ist die Sternhöhe von $T_{d,h}(P)$ höchstens h. Nebenbei sei bemerkt, dass man r nicht in allen Fällen so konstruieren kann, dass $\mathsf{dg}(r) = d$ und $\mathsf{sh}(r) = h$ gilt, z.B. wenn in der induktiven Konstruktion einige der Sprachen $T_{d,h-1}(P_i^{-1}P_i)$ leer sind.

Ausgehend von Lemma 6.3 kann man durch Induktion über h zeigen:

Lemma 6.5. *Für alle $d \geq 1$, $h \in \mathbb{N}$ und $P \subseteq M$ gilt $T_{d,h}(P) \subseteq \eta^{-1}(P) \setminus \varepsilon$.*

Aus Lemma 6.3 und 6.5 folgt somit für alle $h \in \mathbb{N}$ und $P \subseteq M$:

$$\eta^{-1}(P) \setminus \varepsilon \;=\; \bigcup_{d \geq 1} T_{d,0}(P) \;\subseteq\; \bigcup_{d \geq 1} T_{d,h}(P) \;\subseteq\; \eta^{-1}(P) \setminus \varepsilon. \tag{1}$$

Es seien $h \in \mathbb{N}$ und $P \subseteq M$. Wir betrachten die Gleichung (1). Falls für ein $d \geq 1$ $T_{d,h}(P) = \eta^{-1}(P) \setminus \varepsilon$ gilt, dann sagen wir, dass die $T_{d,h}(P)$-Hierarchie *im Level d kollabiert*. Wenn die $T_{d,h}(P)$-Hierarchie im Level d kollabiert, dann gilt (mit Lemma 6.1)

$$\mathsf{sh}\big(\eta^{-1}(P)\big) \;=\; \mathsf{sh}\big(T_{d,h}(P)\big) \;\leq\; h.$$

Wir zeigen nun, dass auch die Umkehrung dieser Aussage gilt.

Lemma 6.6. *Es sei r ein Stringausdruck mit $\varepsilon \notin L(r)$. Weiterhin seien $d \geq \mathsf{dg}(r)$, $h \geq \mathsf{sh}(r)$ und $\eta(L(r)) \subseteq P \subseteq M$. Es gilt $L(r) \subseteq T_{d,h}(P)$.*

Beweisidee. Für $r = \emptyset$ ist die Aussage offensichtlich. Es sei $r \neq \emptyset$. Wir führen den Beweis induktiv über die Sternhöhe von r.

Es sei r ein Stringausdruck mit $\mathsf{sh}(r) = 0$ und d, h, P wie in dem Lemma. Die Sprache $L(r)$ ist dann eine endliche Sprache von Worten w mit $1 \leq |w| \leq d$ und $\eta(w) \in P$. Nach Lemma 6.3 gilt $L(r) \subseteq T_{d,h}(P)$.

Nun sei r ein einzelner Stringausdruck mit $\mathsf{sh}(r) \geq 1$. Es seien d, h und P wie in dem Lemma. Wegen $\mathsf{dg}(r) \leq d$ gibt es ein $1 \leq c \leq d$, $a_1, \ldots, a_c \in \Sigma$ und Stringausdrücke $r_1, \ldots, r_{c-1}$, so dass $r = a_1 r_1^* a_2 r_2^* \ldots r_{c-1}^* a_c$ gilt. Wir können annehmen, dass $\varepsilon \notin L(r_i)$ für alle $1 \leq i < c$ gilt.

Für $1 \leq i < c$ sei $P_i := \eta\big(L(a_1 r_1^* \ldots a_i r_i^*)\big)$ und $P_0 := \{1\}$, $P_c := \eta(L(r))$.

Aus dieser Definition der Mengen P_i kann man für alle $1 \leq i < c$ schlußfolgern, dass $P_{i-1} \cdot \eta(a_i) \subseteq P_i$ gilt, und somit a_i in $T_{1,0}(P_{i-1}^{-1} P_i)$ enthalten ist.

Es sei $1 \leq i < c$ beliebig. Es gilt $P_i \cdot \eta(L(r_i)) \subseteq P_i$ und somit $\eta(L(r_i)) \subseteq P_i^{-1} P_i$. Weil $\mathsf{dg}(r_i) \leq \mathsf{dg}(r) \leq d$ und $\mathsf{sh}(r_i) < \mathsf{sh}(r) \leq h$ gilt, können wir die Induktionsvoraussetzung auf r_i anwenden, und erhalten $L(r_i) \subseteq T_{d,h-1}(P_i^{-1} P_i)$.

Aus der Definition von $T_{d,h}(P)$ folgt mit $P_0, \ldots, P_c$ $L(r) \subseteq T_{d,h}(P)$.

Den Beweis für Stringausdrücke r mit $\mathsf{sh}(r) \geq 1$ führt man durch Zerlegung von r in einzelne Stringausdrücke auf welche man den bereits gezeigten Spezialfall anwendet. □

Es sei $P \subseteq M$. Falls die Sternhöhe von $\eta^{-1}(P)$ kleiner gleich h ist, dann wird nach Lemma 6.1 und 6.2 die Sprache $\eta^{-1}(P) \backslash \varepsilon$ durch einen Stringausdruck r mit $\mathsf{sh}(r) \leq h$ erzeugt. Es sei $d := \mathsf{dg}(r)$. Nach Lemma 6.6 und 6.5 gilt

$$\eta^{-1}(P) \setminus \varepsilon \;=\; L(r) \;\subseteq\; T_{d,h}(P) \;\subseteq\; \eta^{-1}(P) \setminus \varepsilon.$$

Somit kollabiert die $T_{d,h}(P)$-Hierarchie in Level d, und es folgt:

Satz 6.7. *Es seien $h \in \mathbb{N}$ und $P \subseteq M$. Es gilt genau dann $\mathsf{sh}(\eta^{-1}(P)) \leq h$, wenn die $T_{d,h}(P)$-Hierarchie kollabiert, d.h. wenn es ein $d \geq 1$ gibt, so dass $T_{d,h}(P) = \eta^{-1}(P) \setminus \varepsilon$ gilt.*

4.3 Eine Reduktion auf das Beschränktheitsproblem

Um zu ermitteln, ob die Sternhöhe von $\eta^{-1}(P)$ kleiner gleich h ist, beschäftigen wir uns nun mit der Frage, wie man zu gegebenen $h \in \mathbb{N}$, $P \subseteq M$ entscheiden kann, ob die $T_{d,h}(P)$-Hierarchie kollabiert. Hierzu spielen verschachtelte Distanzdesertautomaten eine zentrale Rolle. Wir konstruieren einen $(h+1)$-fach verschachtelten Distanzdesertautomaten $\mathcal{A}_h(P)$, der $\eta^{-1}(P) \setminus \varepsilon$ akzeptiert. Für jedes Wort $w \in \eta^{-1}(P) \setminus \varepsilon$ berechnet der Automat die kleinste Zahl $d \in \mathbb{N}$ für die $w \in T_{d+1,h}(P)$ gilt. Somit ist $\mathcal{A}_h(P)$ genau dann beschränkt, wenn die $T_{d,h}(P)$-Hierarchie kollabiert. Nach Satz 6.7 ist dies genau dann der Fall, wenn die Sternhöhe von $\eta^{-1}(P)$ kleiner gleich h ist.

Satz 6.8. *Es seien $h \in \mathbb{N}$ und $P \subseteq M$. Man kann einen $(h+1)$-fach verschachtelten Distanzdesertautomaten $\mathcal{A}_h(P) = [Q, E, q_I, q_F, \theta]$ konstruieren, der die folgenden Eigenschaften erfüllt:*

1. $E \;\subseteq\; (Q \setminus q_F) \times \Sigma \times (Q \setminus q_I)$
2. $|Q| \;\leq\; 2^{|M|h}\big(|M| + 2\big)$
3. *Jede Transition, die den Initialzustand verlässt, ist mit $\curlyvee_h$ markiert. Alle anderen Transitionen haben Markierungen in $\{\curlyvee_0, \ldots, \curlyvee_{h-1}, \angle_0, \ldots, \angle_h\}$.*

4. Es sei Δ die Abbildung, die $\mathcal{A}_h(P)$ berechnet. Für alle $w \in \Sigma^*$, ist $\Delta(w)$ die kleinste Zahl d für die $w \in T_{d+1,h}(P)$ gilt. Insbesondere gilt genau dann $\Delta(w) < \infty$, wenn $w \in \eta^{-1}(P) \setminus \varepsilon$ ist.

Beweisidee. Für $P = \emptyset$ hat $\mathcal{A}_h(P)$ nur die Zustände q_I und q_F und keine Transitionen. Daher nehmen wir $P \neq \emptyset$ an.

Für $h = 0$ ist der Beweis recht einfach. Zunächst nehmen wir M als Zustandsmenge. Für alle $q \in M$, $a \in \Sigma$ fügen wir eine Transition $(q\,, a\,, q \cdot \eta(a))$ ein. Der Initialzustand sei 1 (das neutrale Element von M) und die Menge der akzeptierenden Zustände sei P. Dann wenden wir die übliche Normalisierung an, d.h. wir führen zwei neue Zustände q_I und q_F ein und fügen neue Transitionen hinzu, so dass der Automat $\eta^{-1}(P) \setminus \varepsilon$ akzeptiert. Alle Transitionen die q_I verlassen, markieren wir mit $\curlyvee_0$, alle anderen Transitionen mit $\angle_0$. Offensichtlich erfüllt der Automat (1)(2)(3).

Wir zeigen (4). Sei $w \in \eta^{-1}(P) \setminus \varepsilon$. Nach der Konstruktion akzeptiert der Automat w. Jeder erfolgreiche Pfad für w ist mit $\curlyvee_0 \angle_0^{|w|-1}$ markiert. Somit gilt $\Delta(w) = |w| - 1$. Nach Lemma 6.3 gilt $w \in T_{|w|,0}$, jedoch $w \notin T_{|w|-1,0}$. Somit ist (4) für alle Worte in $\eta^{-1}(P) \setminus \varepsilon$ erfüllt. Für alle anderen Worte ist (4) auch erfüllt, weil der Automat diese Worte nicht akzeptiert und diese Worte für kein $d \geq 1$ in $T_{d,0}(P)$ enthalten sind.

Nun sei $h \geq 0$ und $P \subseteq M$. Wir konstruieren $\mathcal{A}_{h+1}(P)$. Durch Induktion setzen wir voraus, dass für h und alle Mengen $R \subseteq M$ ein Automat $\mathcal{A}_h(R)$ mit den Eigenschaften (1),...,(4) bereits zur Verfügung steht. Wir konstruieren $\mathcal{A}_{h+1}(P)$ schrittweise.

Zunächst seien alle nichtleeren Teilmengen von M Zustände. Für je zwei Mengen $S, T \in \mathcal{P}_{ne}(M)$ und $a \in \Sigma$ fügen wir eine Transition (S, a, T) ein, falls $S \cdot \eta(a) \subseteq T$ und $S \neq T$ gilt. Die Einschränkung $S \neq T$ ist aus beweistechnischen Gründen notwendig. Diesmal ist $\{1\}$ der Initialzustand und alle nichtleeren Teilmengen von P sind akzeptierende Zustände. Wie oben transformieren wir den Automaten, so dass (1) erfüllt ist. Alle Transitionen die q_I verlassen, markieren wir mit $\curlyvee_{h+1}$, alle anderen Transitionen mit $\angle_{h+1}$. Wir erhalten einen Automaten $\mathcal{A}' := [Q', E', q_I, q_F, \theta']$. Dieser Automat akzeptiert nur Worte in $\eta^{-1}(P) \setminus \varepsilon$, jedoch i.A. nicht alle Worte aus $\eta^{-1}(P) \setminus \varepsilon$, da wir keine Transitionen der Form (S, a, S) zugelassen haben.

Nun führen wir für alle nichtleeren Mengen $S \subseteq M$ folgende Prozedur durch: Wir fügen eine disjunkte Kopie von $\mathcal{A}_h(S^{-1}S)$ in den Automaten $\mathcal{A}'$ ein, und verschmelzen den initialen und den akzeptierenden Zustand von $\mathcal{A}_h(S^{-1}S)$ mit dem Zustand S von $\mathcal{A}'$. Der so erhaltene Automat ist $\mathcal{A}_{h+1}(P)$. Da wir in $\mathcal{A}'$ keine Transitionen der Form (S, a, S) zugelassen haben, ist die Vereinigung der Transitionen disjunkt, so dass die Markierung der Transitionen von $\mathcal{A}'$ bzw. $\mathcal{A}_h(S^{-1}S)$ übernommen werden kann.

Der Automat $\mathcal{A}_{h+1}(P)$ erfüllt (1),(3), und man kann leicht nachrechnen, dass (2) erfüllt ist. Für (4) setzt man induktiv voraus, dass die eingefügten Automaten $\mathcal{A}_h(R)$ (4) erfüllen. □

4.4 Die Entscheidbarkeit und PSPACE-Härte des Sternhöhenproblems

Beweisidee zu Theorem 2.3. Aus den Beweisen von KLEENE's Theorem folgt $\mathsf{sh}(L) \leq n$. Somit kann der Entscheidungsalgorithmus im Fall $h \geq n$ mit "ja" antworten, ohne Berechnungen auszuführen.

Falls $h < n$ gilt, dann konstruiert der Algorithmus das syntaktische Monoid und den syntaktischen Homomorphismus $\eta : \Sigma^* \to M$ von L. Danach konstruiert er den Automaten $\mathcal{A}_h(P)$ in Satz 6.8 für $P := \eta(L)$ und entscheidet mit Theorem 2.2, ob $\mathcal{A}_h(P)$ beschränkt ist. Falls $\mathcal{A}_h(P)$ beschränkt ist, dann antwortet der Algorithmus "ja", ansonsten "nein". Nach Satz 6.8(4), Satz 6.7 und Lemma 6.1 ist $\mathcal{A}_h(P)$ genau dann beschränkt, wenn die Sternhöhe von L kleiner gleich h ist.

Die Kompexitätsschranke folgt aus Satz 6.8(2) und Theorem 2.2. Der Test $h \geq n$ ist nur für die Analyse der Komplexität von Bedeutung und der Algorithmus ist auch ohne diesen Test korrekt. □

Wir zeigen nun, dass das Sternhöhenproblem PSPACE-hart ist. Dazu entwickeln wir eine Reduktion des Universalitätsproblems für nichtdeterministische Automaten auf das Sternhöhenproblem.

Lemma 6.9. *Es seien $\Sigma = \{a, b\}$ und $K, L \subseteq \Sigma^*$ erkennbare Sprachen und $L' := \Sigma^* cL \cup Kc\Sigma^*$.*

1. *Falls $L = \Sigma^*$, dann gilt $\mathsf{sh}(L') = 1$.*

2. *Falls $L \subsetneq \Sigma^*$, dann gilt $\mathsf{sh}(L') \geq \mathsf{sh}(K)$.*

Beweisidee. (1) Es gilt $L' = \Sigma^* c\Sigma^*$ und damit $\mathsf{sh}(L') = 1$.

(2) Es sei r ein Stringausdruck mit $L(r) = L'$ und $\mathsf{sh}(r) = \mathsf{sh}(L')$. Dann ist r eine endliche Vereinigung von einzelnen Stringausdrücken der Form $r_i' c r_i''$, wobei c in r_i' und r_i'' nicht auftritt. Nun seien $w \in \Sigma^* \setminus L$ und s die Vereinigung der Ausdrücke r_i' für alle i mit $w \in L(r_i'')$. Es gilt $L(s) = K$ und $\mathsf{sh}(L') = \mathsf{sh}(r) \geq \mathsf{sh}(s) \geq \mathsf{sh}(K)$. □

Beweisidee für Theorem 2.4. Es sei $K \subseteq \{a, b\}^*$ eine erkennbare Sprache mit $\mathsf{sh}(K) > h$ [6]. Nach Lemma 6.9 gilt für jede erkennbare Sprache $L \subseteq \{a, b\}^*$ genau dann $L = \Sigma^*$, wenn $\mathsf{sh}(\Sigma^* cL \cup Kc\Sigma^*) \leq h$. Da das Problem, ob $L = \Sigma^*$ gilt, PSPACE-vollständig ist, ist das Problem ob die Sternhöhe einer erkennbaren Sprache über $\{a, b, c\}$ kleiner gleich h ist, PSPACE-hart.

Zur Übertragung auf Sprachen über $\{a, b\}$ verwenden wir den durch $\alpha(a) := aa$, $\alpha(b) := ab$ und $\alpha(c) := ba$ definierten Homomorphismus α, der die Sternhöhe erkennbarer Sprachen erhält [19]. □

4.5 Über Sternhöhensubstitutionen

Wir skizzieren den Beweis von Theorem 2.5(2) für den Spezialfall nichtlöschender, nichtleerer Substitutionen. Es seien $K \subseteq (\Sigma \cup X)^*$ und $L \subseteq \Sigma^*$ erkennbare Sprachen, die von nichtdeterministischen Automaten $[Q_K, E_K, I_K, F_K]$ bzw. $[Q_L, E_L, I_L, F_L]$ erkannt werden. Das syntaktische Monoid und den syntaktischen Homomorphismus von L bezeichnen wir mit $\eta : \Sigma^* \to M$. Weiterhin sei $h \geq 0$.

Es sei $\kappa : X \to \mathcal{P}(M)$ eine Abbildung. Durch $\kappa(a) := \eta(a)$ für $a \in \Sigma$, wird κ zu einem Homomorphismus $\kappa : (\mathcal{P}(\Sigma \cup X)^*, \cup, \cdot) \to (\mathcal{P}(M), \cup, \cdot)$ erweitert. Wir bezeichnen κ als einen *Typ*, falls $\kappa(K) = \eta(L)$ gilt und für alle $x \in X$, $\eta^{-1}(\kappa(x))$ mindestens ein nichtleeres Wort enthält.

Es sei $\sigma : X \to \mathcal{P}(\Sigma^*)$ eine Substitution. Für alle $x \in X$ definieren wir $\kappa(x) := \eta(\sigma(x))$. Wie oben erweitern wir κ auf $\mathcal{P}((\Sigma \cup X)^*)$. Für alle $a \in \Sigma$ gilt dann $\kappa(a) = \eta(a) = \eta(\sigma(a))$. Somit gilt $\kappa = \eta \circ \sigma$ und insbesondere $\eta(\sigma(K)) = \kappa(K)$. Falls σ nichtlöschend und nichtleer ist, dann ist κ ein Typ. Es sei κ ein Typ und $d \geq |M|$. Wir betrachten Substitutionen der Form $\sigma(x) := T_{d,h}(\kappa(x))$ für $x \in X$. Wie in Kapitel 4.2 bemerkt, ist σ nichtlöschend und es gilt $\mathsf{sh}(\sigma) \leq h$. Weil κ ein Typ ist, gibt es ein $w \in \Sigma^+$ mit $\eta(w) \in \kappa(x)$. Durch ein Zählargument kann man annehmen, dass $1 \leq |w| \leq |M|$ gilt. Wegen $d \geq |M|$ gilt $w \in T_{d,0}(\kappa(x)) \subseteq T_{d,h}(\kappa(x)) = \sigma(x)$. Somit ist σ nichtleer.

Lemma 7.3. *Die folgenden Aussagen sind äquivalent:*

1. *Es gibt eine nichtlöschende, nichtleere Substitution σ' mit $\sigma'(K) = L$ und $\mathsf{sh}(\sigma') \leq h$.*

2. *Es gibt einen Typ κ und ein $d \geq |M|$, so dass $\sigma(K) = L$ für die Substitution σ die durch $\sigma(x) := T_{d,h}(\kappa(x))$ für $x \in X$ definiert wird.*

Beweisidee. Für (2) $\Rightarrow$ (1) setzt man $\sigma' := \sigma$. Für (1) $\Rightarrow$ (2) setzt man $\kappa(x) := \eta(\sigma'(x))$. □

Es sei κ ein Typ. Wir zeigen, dass es entscheidbar ist, ob (2) in Lemma 7.3 für κ erfüllt ist.

Dazu konstruieren wir einen $(h+1)$-fach verschachtelten Distanzdesertautomaten $\mathcal{A}_\kappa$. Ausgangspunkt der Konstruktion ist $\mathcal{A}_K$. Wir markieren alle Transitionen in $\mathcal{A}_K$ mit Υ_h. Nun ersetzen wir jede Transition (p, x, q) in $\mathcal{A}_K$ durch den Automaten $\mathcal{A}_h(\kappa(x))$ aus Satz 6.8. Wir fügen dazu eine disjunkte Kopie von $\mathcal{A}_h(\kappa(x))$ in $\mathcal{A}_K$ ein und verschmelzen den Initial- und akzeptierenden Zustand von $\mathcal{A}_h(\kappa(x))$ mit p bzw. q. Dadurch erhalten wir einen Automaten, den wir als $\mathcal{A}_\kappa$ bezeichnen.

Beim Einfügen der Automaten $\mathcal{A}_h(\kappa(x))$ in $\mathcal{A}_K$ ist die Vereinigung der Transitionen in manchen Fällen nicht disjunkt. Dies ist bsw. der Fall, wenn es in $\mathcal{A}_K$ Transitionen (p, a, q) und (p, x, q) gibt, und es in $\mathcal{A}_h(\kappa(x))$ eine Transition (q_I, a, q_F) gibt. Man kann jedoch zeigen, dass zusammenfallende Transitionen stets mit Υ_h markiert sind, so dass man durch das Übernehmen der Markierungen von $\mathcal{A}_K$ und den Automaten $\mathcal{A}_h(\kappa(x))$ eine wohldefinierte Markierung der Transitionen von $\mathcal{A}_\kappa$ erhält.

Satz 7.4. *Es sei κ ein Typ. Die folgenden Aussagen sind äquivalent:*

1. *Es gibt ein $d \geq |M|$, so dass $\sigma(K) = L$ für die Substitution σ mit $\sigma(x) := T_{d,h}\big(\kappa(x)\big)$ für alle $x \in X$ gilt.*
2. *Es gilt $L(\mathcal{A}_\kappa) = L$ und $\mathcal{A}_\kappa$ ist beschränkt.*

In (1) $\Rightarrow$ (2) kann man zeigen, dass $\mathcal{A}_\kappa$ mit $d - 1$ beschränkt ist. Bei (2) $\Rightarrow$ (1) kann man $d := \max\{|M|, c\}$ setzen, wobei $c + 1$ die grösste Ausgabe von $\mathcal{A}_\kappa$ auf L ist.

Beweisidee zu Theorem 2.5. In dieser Kurzfassung betrachten wir nur nichtlöschende, nichtleere Substitutionen. Zuerst konstruiert der Algorithmus das syntaktische Monoid und den syntaktischen Homomorphismus $\eta : \Sigma^* \to M$ von L. Danach konstruiert der Algorithmus zu jedem Typ κ den Automaten $\mathcal{A}_\kappa$ und testet, ob Bedingung (2) in Satz 7.4 erfüllt ist, d.h. er testet, ob $L(\mathcal{A}_\kappa) = L$ gilt und ob $\mathcal{A}_\kappa$ beschränkt ist. Wenn dies für einen Typen κ der Fall ist, dann antwortet der Algorithmus mit “ja”, ansonsten mit “nein”. □

5 Schlussfolgerungen und Herausforderungen

Der Autor sieht in diesem Gebiet im wesentlichen zwei Herausforderungen:

Die erste Herausforderung ist eine Ermittlung der genauen Komplexität des Sternhöhenproblems. Es ist völlig offen, ob die Reduktion in Kapitel 4 bzw. in [24] optimiert werden kann.

Die zweite Herausforderung ist eine Verallgemeinerung der Ideen dieser Habilitationsschrift um Entscheidbarkeitsresultate für andere Hierarchien erkennbarer Sprachen zu erhalten, wie z.B. der Straubing-Thérien-Hierarchie, der Dot-Depth-Hierarchie und der berühmten erweiterten Sternhöhenhierarchie.

Darüber hinaus gibt es einige weitere offene Fragen wie z.B. die Entscheidbarkeit des Äquivalenzproblems für Desertautomaten.

Man sollte eine möglichst kleine obere Schranke für die Abbildungen beschränkter Distanzdesertautomaten finden. Selbst für Distanzautomaten ist dieses Problem nicht zufriedenstellend gelöst. Für Distanzautomaten mit n Zuständen ist die niedrigste bekannte obere Schranke $2^{3n^3+n\lg n+n-1}$ [35], aber die extremsten bekannten Beispiele sind mit $2^n - 2$ beschränkt [33, 54].

Weiterhin sollte man nach Zusammenhängen zwischen Distanzdesertautomaten und anderen Konzepten der Theoretischen Informatik suchen und weitere Anwendungsfelder erschließen.

References

[1] S. Bala. Regular language matching and other decidable cases of the satisfiability problem for constraints between regular open terms. In V. Diekert and M. Habib, editors, *STACS'04 Proceedings*, volume 2996 of *Lecture Notes in Computer Science*, pages 596–607. Springer-Verlag, Berlin, 2004.

[2] S. Bala. Complexity of regular language matching and other decidable cases of the satisfiability problem for constraints between regular open terms. *Theory of Computing Systems, special issue of selected best papers from STACS 2004*, 39(1):137 – 163, 2006.

[3] J. Berstel. *Transductions and Context-Free Languages*. B. G. Teubner, Stuttgart, 1979.

[4] R. S. Cohen. Star height of certain families of regular events. *Journal of Computer and System Sciences*, 4:281–297, 1970.

[5] K. Culik II and J. Kari. Image compression using weighted finite automata. *Computer & Graphics*, 17:305–313, 1993.

[6] F. Dejean and M.-P. Schützenberger. On a question of Eggan. *Information and Control*, 9:23–25, 1966.

[7] L. C. Eggan. Transition graphs and the star height of regular events. *Michigan Math. Journal*, 10:385–397, 1963.

[8] S. Eilenberg. *Automata, Languages, and Machines,* Vol. A. Academic Press, New York, 1974.

[9] G. Grahne and A. Thomo. Approximate reasoning in semi-structured databases. In M. Lenzerini et al., editors, *8th International Workshop on Knowledge Representation meets Databases (KRDB2001)*, volume 45 of *CEUR Workshop Proceedings*, 2001.

[10] P. A. Grillet. *Semigroups: An Introduction to the Structure Theory*, volume 193 of *Monographs and Textbooks in Pure and Applied Mathematics*. Marcel Dekker, Inc., New York, 1995.

[11] K. Hashiguchi. Limitedness theorem on finite automata with distance functions. *Journal of Computer and System Sciences*, 24:233–244, 1982.

[12] K. Hashiguchi. Regular languages of star height one. *Information and Control*, 53:199–210, 1982.

[13] K. Hashiguchi. Representation theorems of regular languages. *Journal of Computer and System Sciences*, 27(1):101–115, 1983.

[14] K. Hashiguchi. Algorithms for determining relative star height and star height. *Information and Computation*, 78:124–169, 1988.

[15] K. Hashiguchi. Improved limitedness theorems on finite automata with distance functions. *Theoretical Computer Science*, 72(1):27–38, 1990.

[16] K. Hashiguchi. New upper bounds to the limitedness of distance automata. In F. Meyer auf der Heide and B. Monien, editors, *ICALP'96 Proceedings*, volume 1099 of *Lecture Notes in Computer Science*, pages 324–335. Springer-Verlag, Berlin, 1996.

[17] K. Hashiguchi. New upper bounds to the limitedness of distance automata. *Theoretical Computer Science*, 233:19–32, 2000.

[18] K. Hashiguchi. Erratum to "New upper bounds to the limitedness of distance automata". *Theoretical Computer Science*, 290(3):2183–2184, 2003.

[19] K. Hashiguchi and N. Honda. Homomorphisms that preserve star height. *Information and Control*, 30:247–266, 1976.

[20] J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory Languages, and Computation.* Addison-Wesley, Reading, 1979.

[21] F. Katritzke. *Refinements of Data Compression using Weighted Finite Automata*. PhD thesis, Universität Siegen, 2001.

[22] D. Kirsten. Desert automata and the finite substitution problem. In V. Diekert and M. Habib, editors, *STACS'04 Proceedings*, volume 2996 of *Lecture Notes in Computer Science*, pages 305–316. Springer-Verlag, Berlin, 2004.

[23] D. Kirsten. Distance desert automata and the star height one problem. In I. Walukiewicz, editor, *FoSSaCS'04 Proceedings*, volume 2987 of *Lecture Notes in Computer Science*, pages 257–272. Springer-Verlag, Berlin, 2004.

[24] D. Kirsten. Distance desert automata and the star height problem. *R.A.I.R.O. - Informatique Théorique et Applications, special issue of selected best papers from FoSSaCS 2004*, 29(3):455–509, 2005.

[25] D. Kirsten. A Burnside approach to the finite substitution problem. *Theory of Computing Systems, special issue of selected best papers from STACS 2004*, 39(1):15 – 50, 2006.

[26] D. Kirsten and J. Marcinkowski. Two techniques in the area of the star problem in trace monoids. *Theoretical Computer Science*, 309(1-3):381–412, 2003.

[27] D. Krob. The equality problem for rational series with multiplicities in the tropical semiring is undecidable. *International Journal of Algebra and Computation*, 4(3):405–425, 1994.

[28] G. Lallement. *Semigroups and Combinatorial Applications*. John Wiley & Sons, New York, 1979.

[29] H. Leung. *An Algebraic Method for Solving Decision Problems in Finite Automata Theory*. PhD thesis, Pennsylvania State University, Department of Computer Science, 1987.

[30] H. Leung. On the topological structure of a finitely generated semigroup of matrices. *Semigroup Forum*, 37:273–287, 1988.

[31] H. Leung. Limitedness theorem on finite automata with distance functions: An algebraic proof. *Theoretical Computer Science*, 81(1):137–145, 1991.

[32] H. Leung. On some decision problems in finite automata. In J. Rhodes, editor, *Monoids and Semigroups with Applications*, pages 509–526. World Scientific, Singapore, 1991.

[33] H. Leung. On finite automata with limited nondeterminism. *Acta Informatica*, 35(7):595–624, 1998.

[34] H. Leung. The topological approach to the limitedness problem on distance automata. In J. Gunawardena, editor, *Idempotency*, pages 88–111. Cambridge University Press, 1998.

[35] H. Leung and V. Podolskiy. The limitedness problem on distance automata: Hashiguchi's method revisited. *Theoretical Computer Science*, 310(1-3):147–158, 2004.

[36] S. Lombardy. *Approche structurelle de quelques problèmes de la théorie des automates*. PhD thesis, École nationale supérieure des télécommunications, Paris, 2001.

[37] S. Lombardy and J. Sakarovitch. On the star height of rational languages. A new proof for two old results. In M. Ito, editor, *Proc. of the 3rd Int. Conf. on Languages, Words and Combinatorics, Kyoto'00*. World Scientific, 2000.

[38] S. Lombardy and J. Sakarovitch. Star height of reversible languages and universal automata. In *LATIN'02 Proceedings*, volume 2286 of *Lecture Notes in Computer Science*, pages 76–89. Springer-Verlag, Berlin, 2002.

[39] R. McNaughton. The loop complexity of pure-group events. *Information and Control*, 11:167–176, 1967.

[40] Y. Métivier and G. Richomme. New results on the star problem in trace monoids. *Information and Computation*, 119(2):240–251, 1995.

[41] M. Mohri. Finite-state transducers in language and speech processing. *Computational Linguistics*, 23:269–311, 1997.

[42] R. Montalbano and A. Restivo. On the star height of rational languages. *International Journal of Algebra and Computation*, 4(3):427–441, 1994.

[43] D. Perrin. Finite automata. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, pages 1–57. Elsevier Science Publishers, 1990.

[44] J.-É. Pin. *Varieties of Formal Languages*. North Oxford Academic Publishers Ltd, 1986.

[45] J.-É. Pin. Rational and recognizable langages. In Ricciardi, editor, *Lectures in Applied Mathematics and Informatics*, pages 62–106. Manchester University Press, 1990.

[46] J.-É. Pin. Finite semigroups and recognizable languages: An introduction. In J. Fountain, editor, *NATO Advanced Study Institute, Semigroups, Formal Languages and Groups*, pages 1–32. Kluwer Academic Publishers, 1995.

[47] J.-É. Pin. Syntactic semigroups. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages, Vol. 1, Word, Language, Grammar*, pages 679–746. Springer-Verlag, Berlin, 1997.

[48] J.-É. Pin. Tropical semirings. In J. Gunawardena, editor, *Idempotency*, pages 50–69. Cambridge University Press, 1998.

[49] J.-É. Pin. Personal communication, 2003.

[50] I. Simon. Limited subsets of a free monoid. In *Proceedings of the 19th IEEE Annual Symposium on Foundations of Computer Science*, pages 143–150. IEEE Computer Society Press, Long Beach, CA, 1978.

[51] I. Simon. Recognizable sets with multiplicities in the tropical semiring. In M. P. Chytil et al., editors, *MFCS'88 Proceedings*, volume 324 of *Lecture Notes in Computer Science*, pages 107–120. Springer-Verlag, Berlin, 1988.

[52] I. Simon. The nondeterministic complexity of a finite automaton. In M. Lothaire, editor, *Mots - mélanges offerts à M.-P. Schützenberger*, pages 384–400. Hermes, 1990.

[53] I. Simon. On semigroups of matrices over the tropical semiring. *R.A.I.R.O. - Informatique Théorique et Applications*, 28:277–294, 1994.

[54] A. Weber. Distance automata having large finite distance or finite ambiguity. *Mathematical Systems Theory*, 26:169–185, 1993.

[55] A. Weber. Finite valued distance automata. *Theoretical Computer Science*, 134:225–251, 1994.

[56] S. Yu. Regular Languages. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages, Vol. 1, Word, Language, Grammar*, pages 41–110. Springer-Verlag, Berlin, 1997.

Selbständigkeitserklärung

Hiermit erkläre ich, die vorliegende Habilitationsschrift selbständig und ohne unzulässige fremde Hilfe angefertigt zu haben. Ich habe keine anderen als die angeführten Quellen und Hilfsmittel benutzt und sämtliche Textstellen, die wörtlich oder sinngemäß aus veröffentlichten oder unveröffentlichten Schriften entnommen wurden, und alle Angaben, die auf mündlichen Auskünften beruhen, als solche kenntlich gemacht. Ebenfalls sind alle von anderen Personen bereitgestellten Materialien oder erbrachten Dienstleistungen als solche gekennzeichnet.

Dresden, den 1. März 2006

Dr. Daniel Kirsten