Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

Structurally Restricted Weighted Automata

Bachelor Thesis

2022
Pavol Kebis

Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

Structurally Restricted Weighted Automata

Bachelor Thesis

Study Programme:	Computer Science
Field of Study:	Computer Science
Department:	Department of Computer Science
Supervisor:	RNDr. Peter Kostolányi, PhD.

Bratislava, 2022

Pavol Kebis

20167206

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Pavol Kebis
Študijný program: informatika (Jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor: informatika
Typ záverečnej práce: bakalárska
Jazyk záverečnej práce: anglický
Sekundárny jazyk: slovenský

Názov: Structurally Restricted Weighted Automata
Štrukturálne obmedzenia automatov s váhami

Anotácia: Práca skúma základné vlastnosti automatov s váhami nad polokruhmi, ktorých grafy neobsahujú iné orientované kružnice, než slučky (t.j. acycklických automatov s váhami so slučkami). Dôraz sa kladie na prípad automatov nad tropickými polokruhmi.

Vedúci: RNDr. Peter Kostolányi, PhD.
Katedra: FMFI.KI - Katedra informatiky
Vedúci katedry: prof. RNDr. Martin Škoviera, PhD.

Dátum zadania: 18.10.2021

Dátum schválenia: 04.11.2021

doc. RNDr. Daniel Olejár, PhD.
garant študijného programu

....
študent vedúci práce

20167206

Comenius University Bratislava
Faculty of Mathematics, Physics and Informatics

THESIS ASSIGNMENT

Name and Surname:	Pavol Kebis
Study programme:	Computer Science (Single degree study, bachelor I. deg., full time form)
Field of Study:	Computer Science
Type of Thesis:	Bachelor´s thesis
Language of Thesis:	English
Secondary language:	Slovak

Title: Structurally Restricted Weighted Automata

Annotation: The thesis explores fundamental properties of weighted automata over semirings, whose graphs do not contain cycles other than self-loops (i.e., acyclic weighted automata with loops), with emphasis on the case of automata over tropical semirings.

Supervisor:	RNDr. Peter Kostolányi, PhD.
Department:	FMFI.KI - Department of Computer Science
Head of department:	prof. RNDr. Martin Škoviera, PhD.
Assigned:	18.10.2021
Approved:	04.11.2021

doc. RNDr. Daniel Olejár, PhD.
Guarantor of Study Programme

.... Student

.... Supervisor

Acknowledgements: I would like to thank my supervisor, RNDr. Peter Kostolányi, PhD., for his help, guidance, patience and extensive feedback.

Abstrakt

V práci skúmame základné vlastnosti rôznych variantov štrukturálne obmedzených automatov s váhami s dôrazom na prípad automatov nad tropickými polokruhmi. Zaoberáme sa rozhodovacími problémami, ktoré uvažovali S. Almagor, U. Boker a O. Kupfermanová pre všeobecné automaty nad tropickými polokruhmi a skúmame časovú zložitosť a rozhodnuteľnosť pre nasledovné obmedzenia. Definujeme si triedu nerozvetvených automatov, ktoré tvoria podtriedu acyklických automatov so slučkami. Pre nerozvetvené automaty nad tropickým polokruhom prirodzených čísel ukážeme, že problém hornej ohraničenosti a problém absolútnej ohraničenosti patria do triedy **P**, problém konečnosti a problém rovnosti váh všetkých slov patria do **co-NP** a problém existencie slova s danou váhou je **NP**-ťažký. Pre obmedzenie acyklických automatov so slučkami ukážeme, že žiaden z uvažovaných problémov, ktorý bol rozhodnuteľný pre triedu všeobecných automatov s váhami, sa nestane ľahším.

Kľúčové slová: formálny mocninový rad, automat s váhami, acyklický automat so slučkami, rozhodovací problém, tropický polokruh

Abstract

We explore fundamental properties of several variants of structurally restricted weighted automata with emphasis on the case of automata over tropical semirings. We also study hardness of decision problems, considered by S. Almagor, U. Boker, and O. Kupferman for general weighted automata over tropical semirings, in the following restricted settings. We define branchless automata as a subclass of acyclic automata with loops. For branchless automata over tropical semiring of natural numbers we show that the upper boundedness problem and the absolute boundedness problem belong to **P**, the universality problem and the ∀-exact problem belong to **co**-**NP**, and the ∃-exact problem is **NP**-hard. For acyclic weighted automata with loops, we observe that all decision problems that are decidable in the general setting, do not become easier.

Keywords: formal power series, weighted automaton, acyclic automaton with loops, decision problem, tropical semiring

Contents

Notations

In this thesis, we use the following notations:

- $\mathbb{N}$ denotes the set of all natural numbers including 0.
- $\mathbb{N}^+$ denotes the set of all positive natural numbers.
- $[\![a, b]\!]$ denotes the set $\{i \in \mathbb{Z} \mid a \leq i \leq b\}$.
- $[a, b]$ denotes the set $\{i \in \mathbb{R} \mid a \leq i \leq b\}$.
- $|w|$ is the length of a word w.
- $\mathsf{alph}(w)$ is the set of letters occurring in a word w.
- $\#_c(w)$ is the number of occurrences of a letter c in a word w.
- ε denotes the empty word.
- $\emptyset$ denotes the empty set.
- 2^X denotes the set of all subsets of X.
- $A \subseteq B$ denotes that A is a subset of B.
- $A \subsetneq B$ denotes that A is a strict subset of B.
- $A - B$ is the difference of sets A and B.

Introduction

In the early nineties, D. Krob proved that the equality problem for rational series over the tropical semiring of integers is undecidable [12], which was a starting point in the study of decision problems for weighted automata over tropical semirings.

The proof of this result was later simplified by T. Colcombet (unpublished; the proof can be found in [7]) and independently by S. Almagor, U. Boker, and O. Kupferman [1]. The latter authors have actually obtained a stronger result, proving undecidability of the universality problem for tropical weighted automata. They also initiated a systematic study of several related decision problems, which they later deepened in [2].

Many problems studied in [2] turned out to be hard or even undecidable, which raises the question of whether these problems would become easier for some restricted class of weighted automata over tropical semirings. The case of deterministic automata was already studied in [2] where it is shown that most of the problems are decidable in polynomial time.

On the other hand, in this thesis, we focus on structural restrictions imposed on transition graphs of automata.

One of the restricted classes of automata that we study in this thesis consists of acyclic automata with loops, that is, automata without cycles other than loops. In the unweighted context, these automata are also known as acyclic automata [11], partially ordered automata [13] or extensive automata [16]. We use the term acyclic weighted automata *with loops* to distinguish the class from the one considered by M. Mohri [15].

The class of languages realised by acyclic automata with loops coincides with the level 3/2 of the Straubing-Thérien hierarchy [3] and was also studied under the name alphabetic pattern constraints [4]. In the deterministic variant, acyclic automata with loops realise the class of R-trivial languages [5].

Our aim is to explore the basic properties of acyclic *weighted* automata with loops over general semirings and to study decision problems for acyclic weighted automata with loops in the context of tropical semirings. As we demonstrate later, the complexity of the problems decidable for general weighted automata tends not to become easier under the restriction to acyclic automata with loops. This finding led us to focus on a more restricted class of automata, which we call *homogeneous branchless automata*. These automata can be seen as consisting of a single branch of some acyclic automaton with loops, which uses the same letter on each transition between two different states.

For this class of automata over tropical semirings, we obtain new results for the upper boundedness problem, the universality problem and the ∃-exact problem of [2].

Chapter 1

Preliminaries and Known Results

This chapter defines and explains fundamental notions and concepts used throughout the thesis and reviews the previous results in the area of decision problems over tropical semirings. We assume that the reader is familiar with the concepts of finite automata and formal languages.

1.1 Semirings

A *semiring* is an algebra $(S, +, \cdot, 0, 1)$ where S is a set, $+$ is an associative and commutative binary operation over S with 0 as the neutral element, $\cdot$ is an associative binary operation over S with 1 as the neutral element, the operation $\cdot$ distributes over $+$ from left and right, and $a \cdot 0 = 0 \cdot a = 0$ holds for all $a \in S$. The first two conditions can be equivalently formulated as $(S, +, 0)$ being a commutative monoid and $(S, \cdot, 1)$ being a monoid.

Most notable semirings used in connection with weighted automata include:

- The standard semiring of natural numbers $(\mathbb{N}, +, \cdot, 0, 1)$ with the usual addition and multiplication.
- The Boolean semiring $(\mathbb{B}, \vee, \wedge, 0, 1)$.[1]
- The tropical semiring, also known as min-plus semiring, over natural numbers $\mathbb{N}_{min} = (\mathbb{N} \cup \{\infty\}, \min, +, \infty, 0)$ or over integers $\mathbb{Z}_{min} = (\mathbb{Z} \cup \{\infty\}, \min, +, \infty, 0)$.
- The arctic semiring, also known as max-plus semiring, over natural numbers $(\mathbb{N} \cup \{-\infty\}, \max, +, -\infty, 0)$ or over integers $(\mathbb{Z} \cup \{-\infty\}, \max, +, -\infty, 0)$.

[1] An ordinary nondeterministic automaton without weights can be viewed as a weighted automaton over the Boolean semiring.

- The probability semiring over positive real numbers $(\mathbb{R}_+ \cup \{\infty\}, +, \cdot, 0, 1)$.
- The semiring of formal languages $(2^{\Sigma^*}, \cup, \cdot, \emptyset, \{\varepsilon\})$ with the operations of union and concatenation over the set of subsets of language Σ^*.
- Every bounded distributive lattice is a semiring [6].
- Every field is a semiring.

1.2 Formal Power Series

Before we discuss weighted automata, we first need to take a look at formal power series, i.e., the objects realised by weighted automata. Formal power series map words over an alphabet Σ to elements of a semiring S.

In this and the following sections, we assume S to be an arbitrary semiring and Σ to be an arbitrary alphabet.

Definition 1.2.1. A *formal power series* in several noncommutative variables from Σ and with coefficients in S, or a formal power series over S and Σ for short, is a mapping

$$r : \Sigma^* \to S,$$

which is usually written as

$$r = \sum_{w \in \Sigma^*} (r, w)w,$$

where (r, w) is the value of a word w under the mapping r, called the *coefficient* of the word w in the series r. The coefficient (r, ε) is called the *constant coefficient.*

For a power series consisting of one term with coefficient equal to 1 or a series where coefficients of words other than ε are zero, we use a shorter notation, writing only the word from Σ^* or the coefficient from S, respectively.

The *support* of a formal power series r over S and Σ is defined to be a language $\mathrm{supp}(r) = \{w \in \Sigma^* \mid (r, w) \neq 0\}$.

Notation 1.2.2. The set of all formal power series over S and Σ is denoted by $S\langle\langle\Sigma^*\rangle\rangle$. The set of all formal power series over S and Σ whose support is finite set, also called *polynomials* over S and Σ, is denoted by $S\langle\Sigma^*\rangle$. For a given language $L \subseteq \Sigma^*$ we denote by $S\langle\langle L\rangle\rangle$ the set of formal power series $\{r \in S\langle\langle\Sigma^*\rangle\rangle \mid \mathrm{supp}(r) \subseteq L\}$ and similarly $S\langle L\rangle = \{r \in S\langle\Sigma^*\rangle \mid \mathrm{supp}(r) \subseteq L\}$.

Definition 1.2.3. Let $r, s \in S\langle\langle\Sigma^*\rangle\rangle$ be formal power series over S and Σ. The *sum* $r + s$ of series r, s is a formal power series over S and Σ, where for every word w from Σ^*,

$$(r + s, w) = (r, w) + (s, w).$$

The *Cauchy product* $r \cdot s$ of series r, s is a formal power series over S and Σ, where for every w from Σ^*,

$$(r \cdot s, w) = \sum_{\substack{u,v \in \Sigma^* \\ uv = w}} (r, u) \cdot (s, v).$$

It is known that $(S\langle\langle\Sigma^*\rangle\rangle, +, \cdot, 0, 1)$ with the above-defined sum and Cauchy product is a semiring [6]. The problem that may arise is the impossibility of summing through infinite collections of power series. Thus, we define locally finite family of power series, which ensures that for each word from Σ^*, there is only a finite number of nonzero terms to sum up.

Definition 1.2.4. The family of formal power series $(r_i \mid i \in I)$ over S and Σ is *locally finite* if for every word w from Σ^*, the set $I(w) = \{i \in I \mid (r_i, w) \neq 0\}$ is finite.

Then, the sum $r = \sum_{i \in I} r_i$ of the locally finite family of formal power series $(r_i \mid i \in I)$ is defined for every word w from Σ^* by

$$(r, w) = \sum_{i \in I(w)} (r_i, w).$$

Definition 1.2.5. Let r be a formal power series from $S\langle\langle\Sigma^+\rangle\rangle$. The *star* of the series r is defined as

$$r^* = \sum_{t \in \mathbb{N}} r^t.$$

The family of power series $(r^t \mid t \in \mathbb{N})$ is locally finite because the constant coefficient here is 0 [6].

1.3 Weighted Automata

Weighted finite automata, or simply weighted automata for short,[2] form a generalisation of classical finite automata where every transition has its own weight. These weights are multiplied throughout the run on a given word. Values of particular runs on a given word are then added up.

When not stated otherwise, weighted automata are always assumed to be nondeterministic, i.e., multiple transitions upon a single letter are allowed from a single state. Later

[2] In this thesis, we do not consider infinite automata.

in this section, we also define a deterministic variant of weighted automata, which is weaker than the nondeterministic variant in general. We do not consider transitions on ε, which solves many problems connected with infinite sums over the underlying semiring.

Definition 1.3.1. A *weighted automaton* over S and Σ is a 4-tuple $\mathcal{A} = (Q, \sigma, \iota, \tau)$ where Q is a finite set of states, $\sigma : Q \times \Sigma \times Q \to S$ is a transition weighting function, $\iota : Q \to S$ is an initial weighting function, and $\tau : Q \to S$ is a terminal weighting function.

A *transition* of a weighted automaton $\mathcal{A}$ is a triple $(q, a, p) \in Q \times \Sigma \times Q$ such that $\sigma(q, a, p) \neq 0$.

A *run* from state q_1 to state q_m of an automaton $\mathcal{A} = (Q, \sigma, \iota, \tau)$ is an alternating sequence

$$\gamma = (q_1, a_1, q_2, \ldots, q_{m-1}, a_{m-1}, q_m)$$

of states of the automaton $\mathcal{A}$ and letters from alphabet Σ, where $m \in \mathbb{N}^+$, $q_1, \ldots, q_m \in Q$, $a_1, \ldots a_{m-1} \in \Sigma$ and (q_i, a_i, q_{i+1}) is a transition for every $i \in [\![1, m-1]\!]$.

A run γ from state q_1 to state q_m of an automaton $\mathcal{A} = (Q, \sigma, \iota, \tau)$ is called *successful* if $\iota(q_1) \neq 0$ and $\tau(q_m) \neq 0$.

Let $\mathcal{A} = (Q, \sigma, \iota, \tau)$ be a weighted automaton. The word read during the run $\gamma = (q_1, a_1, q_2, \ldots, q_{m-1}, a_{m-1}, q_m)$ is defined by $\lambda(\gamma) = a_1 \ldots a_{m-1}$ and the length of the run is given by $|\gamma| = m - 1$. We say that γ is a run on $\lambda(\gamma)$. We also use the symbol σ for representing the weight of the run without the initial and the terminal weights, which is given by $\sigma(\gamma) = \prod_{i=1}^{m-1} \sigma(q_i, a_i, q_{i+1})$. Let us also define $\mathcal{R}(\mathcal{A})$ to be the set of all successful runs of the automaton $\mathcal{A}$.

For a given automaton $\mathcal{A} = (Q, \sigma, \iota, \tau)$, states $p, q \in Q$, and word $w \in \Sigma^*$, we write $p \xrightarrow{w} q$ if there exists a run on w from p to q in $\mathcal{A}$.

Remark 1.3.2. Without loss of generality, we often confine ourselves to automata where the set of states is $[\![1, n]\!]$ for some $n \in \mathbb{N}^+$, i.e., $\mathcal{A} = ([\![1, n]\!], \sigma, \iota, \tau)$. Such automaton is denoted as $\mathcal{A} = (n, \sigma, \iota, \tau)$.

In the case of classical finite automata, an automaton realises a formal language. In the case of weighted automata, we need to define the formal power series realised by a given automaton $\mathcal{A}$, which is denoted by $||\mathcal{A}||$.

The monomial realised by a run γ of an automaton $\mathcal{A}$ from state p to q is

$$||\gamma|| = (\iota(p)\sigma(\gamma)\tau(q))\lambda(\gamma).$$

Definition 1.3.3. Let $\mathcal{A}$ be a weighted automaton over S and Σ. A series realised by the automaton $\mathcal{A}$ is defined by

$$||\mathcal{A}|| = \sum_{\gamma \in \mathcal{R}(\mathcal{A})} ||\gamma||.$$

Since we do not consider transitions on ε, the family of formal power series $(||\gamma|| \mid \gamma \in \mathcal{R}(\mathcal{A}))$ used in the definition of $||\mathcal{A}||$ is locally finite since there is only a finite number of runs on a given word.

Definition 1.3.4. A *deterministic weighted automaton* over S and Σ is a weighted automaton $\mathcal{A} = (Q, \sigma, \iota, \tau)$ such that

(i) For each states $p, q_1, q_2 \in Q$ and letter $a \in \Sigma$, $q_1 = q_2$ holds if $\sigma(p, a, q_1) \neq 0$ and $\sigma(p, a, q_2) \neq 0$.

(ii) There is no more than one state $p \in Q$ which has nonzero initial weight, i.e., $\iota(p) \neq 0$.

Definition 1.3.5. Let $\mathcal{A} = (Q, \sigma, \iota, \tau)$ be a weighted automaton over S and Σ. A state $p \in Q$ is *useful* if there exists a successful run in $\mathcal{A}$ which passes through p. The automaton $\mathcal{A}$ is called *trim* if every state $p \in Q$ is useful.

See, e.g., J. Sakarovitch [17], for a proof of the following simple propositions for unweighted finite automata. Since the weights do not affect whether a state is useful, the same holds also in a weighted context.

Proposition 1.3.6. For every weighted automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ there exists a weighted trim automaton $\mathcal{A}' = (n', \sigma', \iota', \tau')$ such that $||\mathcal{A}'|| = ||\mathcal{A}||$.

Later on in this thesis, we assume all given automata are trim.

For every weighted automaton, we can also find an equivalent automaton, which has only one state with a nonzero initial weight.

Proposition 1.3.7. [17] For a weighted automaton $\mathcal{A}$ over S and Σ, there exists a weighted automaton $\mathcal{A}' = (Q', \sigma', \iota', \tau')$ over S and Σ and a state $q \in Q'$ such that $\iota'(q) \neq 0$ and $\iota'(p) = 0$ holds for all states $p \in Q' - \{q\}$, and $||\mathcal{A}'|| = ||\mathcal{A}||$.

Definition 1.3.8. Let S be a semiring and Σ an alphabet. For every weighted automaton $\mathcal{A} = (n_{\mathcal{A}}, \sigma_{\mathcal{A}}, \iota_{\mathcal{A}}, \tau_{\mathcal{A}})$ over S and Σ and every weighted automaton $\mathcal{B} = (n_{\mathcal{B}}, \sigma_{\mathcal{B}}, \iota_{\mathcal{B}}, \tau_{\mathcal{B}})$ over S and Σ, *a disjoint union* of $\mathcal{A}$ and $\mathcal{B}$ is a weighted automaton

$\mathcal{C} = (n_\mathcal{C}, \sigma_\mathcal{C}, \iota_\mathcal{C}, \tau_\mathcal{C})$ over S and Σ, denoted by $\mathcal{C} = \mathcal{A} + \mathcal{B}$, such that

$$\begin{aligned} n_\mathcal{C} &= n_\mathcal{A} + n_\mathcal{B} \\ \iota_\mathcal{C}(p) &= \iota_\mathcal{A}(p), \tau_\mathcal{C}(p) = \tau_\mathcal{A}(p) \text{ for every } p \in [1, n_\mathcal{A}] \\ \sigma_\mathcal{C}(p, c, q) &= \sigma_\mathcal{A}(p, c, q) \text{ for every } p, q \in [1, n_\mathcal{A}] \text{ and every } c \in \Sigma \\ \iota_\mathcal{C}(p + n_\mathcal{A}) &= \iota_\mathcal{B}(p), \tau_\mathcal{C}(p + n_\mathcal{A}) = \tau_\mathcal{B}(p) \text{ for every } p \in [1, n_\mathcal{B}] \\ \sigma_\mathcal{C}(p + n_\mathcal{A}, c, q + n_\mathcal{A}) &= \sigma_\mathcal{B}(p, c, q) \text{ for every } p, q \in [1, n_\mathcal{B}] \text{ and every } c \in \Sigma \end{aligned}$$

and every other value of $\sigma_\mathcal{C}, \iota_\mathcal{C}$ and $\tau_\mathcal{C}$ is equal to 0.

Proposition 1.3.9. [18] Let S be a semiring and Σ an alphabet. For every weighted automaton $\mathcal{A} = (n_\mathcal{A}, \sigma_\mathcal{A}, \iota_\mathcal{A}, \tau_\mathcal{A})$ over S and Σ and every weighted automaton $\mathcal{B} = (n_\mathcal{B}, \sigma_\mathcal{B}, \iota_\mathcal{B}, \tau_\mathcal{B})$ over S and Σ, $||\mathcal{A} + \mathcal{B}|| = ||\mathcal{A}|| + ||\mathcal{B}||$.

1.4 Rational Expressions and Rational Series

Another way of realising formal power series is through rational expressions, which form a language over the alphabet $\{\mathsf{r} \mid r \in S\langle\Sigma \cup \{\varepsilon\}\rangle\} \cup \{+, \cdot, ^*, (,)\}$. When defining semantics of rational expressions, we have to avoid using a star operation on expressions which evaluate to series with nonzero constant coefficient.

Definition 1.4.1. A *rational expression* over S and Σ is a word over the alphabet $\{\mathsf{r} \mid r \in S\langle\Sigma \cup \{\varepsilon\}\rangle\} \cup \{+, \cdot, ^*, (,)\}$ defined as follows:

(i) For every series r from $S\langle\Sigma \cup \{\varepsilon\}\}\rangle$ the expression r is a rational expression.

(ii) For every two rational expressions E and F, the expressions $(\mathsf{E} + \mathsf{F})$, $(\mathsf{E} \cdot \mathsf{F})$, and $(\mathsf{E})^*$ are also rational expressions.

(iii) Nothing else is a rational expression.

Next, we define the formal power series $||\mathsf{E}||$ realised by a rational expression E.

Definition 1.4.2. Let E be a rational expression over S and Σ. A formal power series realised by the rational expression E over S and Σ is defined as follows:

(i) If E is a rational expression for which $\mathsf{E} = \mathsf{r}$ for some $r \in S\langle\Sigma \cup \{\varepsilon\}\rangle\}$, then r is the formal power series realised by E.

(ii) If $\mathsf{E} = (\mathsf{F} + \mathsf{G})$ or $\mathsf{E} = (\mathsf{F} \cdot \mathsf{G})$ for some rational expressions F and G, then $||\mathsf{E}|| = ||\mathsf{F}|| + ||\mathsf{G}||$ or $||\mathsf{E}|| = ||\mathsf{F}|| \cdot ||\mathsf{G}||$ respectively.

(iii) If $\mathsf{E} = (\mathsf{F})^*$ for some rational expression F and if there exists a formal power series $||\mathsf{F}||$ with zero constant coefficient, then $||\mathsf{E}|| = ||\mathsf{F}||^*$.

(iv) Other rational expressions do not realise any formal power series over S and Σ.

We say that a rational expression E is valid if there exists a rational series which is realised by E.

Theorem 1.4.3. [17] For a given formal power series r over S and Σ the following statements are equivalent:

(i) There exists a rational expression G such that $||\mathsf{G}|| = r$.

(ii) There exists a weighted automaton $\mathcal{A}$ such that $||\mathcal{A}|| = r$.

Formal power series realised by rational expressions or weighted automata are called *rational series*. The set of all rational series over S and Σ is denoted by $\mathsf{S}\text{-Rat}(\Sigma^*)$.

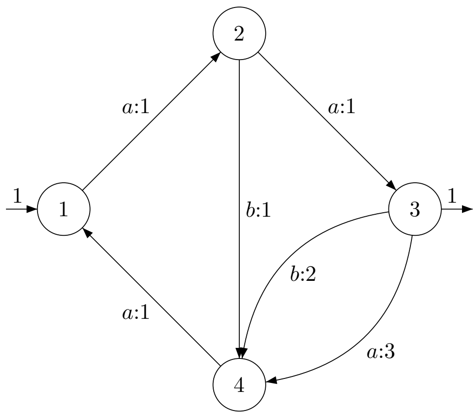

Figure 1.1: Weighted automaton $\mathcal{A}$ over semiring $(\mathbb{N}, +, \cdot, 0, 1)$ and alphabet $\{a, b\}$.

Example 1.4.4. Figure 1.1 shows automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ over semiring $(\mathbb{N}, +, \cdot, 0, 1)$ and the alphabet $\{a, b\}$. Based on the diagram, we deduct $n = 4$, $\sigma(1, a, 2) = 1$, $\sigma(2, a, 3) = 1$, $\sigma(2, b, 4) = 1$, $\sigma(3, a, 4) = 3$, $\sigma(3, b, 4) = 2$, $\sigma(4, a, 1) = 1$, other values of σ are zero, and initial and terminal weights are $\iota(1) = \tau(3) = 1, \iota(2) = \iota(3) = \iota(4) = \tau(1) = \tau(2) = \tau(4) = 0$. We easily see that the automaton $\mathcal{A}$ realises the same series as the rational expression

$$||\mathcal{A}|| = (aba + 2aaba + 3aaaa)^* aa.$$

1.5 Decision Problems over Tropical Semirings

In this section, we present an overview of recent results related to decidability questions for weighted automata over tropical semirings, sometimes referred to as the min-plus semirings.

Weighted automata over tropical semirings assign to each word w the minimum value of a run on w. From various tropical semirings, we consider those used by S. Almagor, U. Boker, and O. Kupferman [2], i.e., $\mathbb{Z}_{min}$ and $\mathbb{N}_{min}$.

Now, we define decision problems for weighted automata that have been considered in [2]. All automata are understood to be over an alphabet Σ and a tropical semiring S such that $S = \mathbb{N}_{min}$ or $S = \mathbb{Z}_{min}$ in what follows.

- *The nonemptiness problem*: Given an automaton $\mathcal{A}$ and a threshold $T \in S$, decide whether there exists a word $w \in \Sigma^*$ such that $(||\mathcal{A}||, w) < T$.
- *The universality problem*: Given an automaton $\mathcal{A}$ and a threshold $T \in S$, decide whether $(||\mathcal{A}||, w) < T$ holds for all words $w \in \Sigma^*$.
- *The upper boundedness problem*: Given an automaton $\mathcal{A}$, decide whether there exists a bound $B \in S$ such that $(||\mathcal{A}||, w) < B$ holds for all words $w \in \Sigma^*$.
- *The absolute boundedness problem*: Given an automaton $\mathcal{A}$, decide whether there exists a bound $B \in S$ such that $-B < (||\mathcal{A}||, w) < B$ holds for all words $w \in \Sigma^*$.
- *The $\exists$-exact problem*: Given an automaton $\mathcal{A}$ and a value $V \in S$, decide whether there exists a word $w \in \Sigma^*$ such that $(||\mathcal{A}||, w) = V$.
- *The $\forall$-exact problem*: Given an automaton $\mathcal{A}$ and a value $V \in S$, decide whether $(||\mathcal{A}||, w) = V$ holds for all words $w \in \mathrm{supp}(\mathcal{A})$.[3]
- *The equality problem*: Given automata $\mathcal{A}$ and $\mathcal{B}$, decide whether $(||\mathcal{A}||, w) = (||\mathcal{B}||, w)$ holds for all words $w \in \Sigma^*$.
- *The containment problem*: Given automata $\mathcal{A}$ and $\mathcal{B}$, decide whether $(||\mathcal{A}||, w) \geq (||\mathcal{B}||, w)$ holds for all words $w \in \Sigma^*$, i.e., whether $||\mathcal{A}||$ contains $||\mathcal{B}||$.

S. Almagor, U. Boker, and O. Kupferman [1, 2] systematically examined the decidability of the decision problems listed above for weighted automata over tropical semirings. They also showed a reduction of the undecidable halting problem for two-counter machines to the universality problem for weighted automata over $\mathbb{Z}_{min}$.[4] They consider

[3] This is the only problem considered in [2] where the set of words is limited to the support of $\mathcal{A}$.
[4] A similar reduction was also obtained by T. Colcombet [7]

decision problems over both deterministic and nondeterministic automata, and both over $\mathbb{Z}_{min}$ and $\mathbb{N}_{min}$. The summary of the decidability results and time complexities is shown in tables 1.1 and 1.2.

	Deterministic $\mathbb{N}_{min}$, $\mathbb{Z}_{min}$	Nondeterministic $\mathbb{N}_{min}$	Nondeterministic $\mathbb{Z}_{min}$
Nonemptiness	**P**	**P**	
Universality	**P**	**PSPACE**-complete	Undecidable
Upper Boundedness	**P**	**PSPACE**-complete	Undecidable
Absolute Boundedness	**P**	**PSPACE**-complete	
$\forall$-exact	**P**	**PSPACE**-complete	
$\exists$-exact	**NP**-complete	**PSPACE**-complete	Undecidable

Table 1.1: Decidability results and time complexities of selected problems examined by S. Almagor, U. Boker, and O. Kupferman [2].

$\|\|\mathcal{A}\|\|$ is equal to $\|\|\mathcal{B}\|\|$	$\mathcal{B}$ is deterministic	$\mathcal{B}$ is nondeterministic
$\mathcal{A}$ is deterministic	**P**	**PSPACE**-complete
$\mathcal{A}$ is nondeterministic	**PSPACE**-complete	Undecidable
$\|\|\mathcal{A}\|\|$ contains $\|\|\mathcal{B}\|\|$	$\mathcal{B}$ is deterministic	$\mathcal{B}$ is nondeterministic
$\mathcal{A}$ is deterministic	**P**	Undecidable
$\mathcal{A}$ is nondeterministic	**P**	Undecidable

Table 1.2: Decidability results and time complexities of the equality and the containment problems examined in [2] for semirings $\mathbb{N}_{min}$ and $\mathbb{Z}_{min}$.

The authors of [2] also showed polynomial reductions between some problems.

Proposition 1.5.1. The following holds for both deterministic and nondeterministic weighted automata over $\mathbb{Z}_{min}$ and $\mathbb{N}_{min}$:

(i) The absolute boundedness problem is polynomially reducible to the upper boundedness problem. [2, p. 18]

(ii) The $\forall$-exact problem is polynomially reducible to the universality problem. [2, p. 16]

(iii) The containment problem for an automaton[5] and a nondeterministic automaton is polynomially reducible to the equality problem for two nondeterministic automata. [2, p. 10]

[5] The reduction is independent of whether the automaton is deterministic or nondeterministic.

(iv) The equality problem for two nondeterministic or two deterministic automata is polynomially reducible to the containment problem for two nondeterministic or two deterministic automata, respectively.

(v) The universality problem for a nondeterministic or deterministic automaton is polynomially reducible to the containment problem of two nondeterministic or deterministic automata respectively.

Proposition 1.5.2. For both deterministic and nondeterministic weighted automata over $\mathbb{N}_{min}$, the upper boundedness problem is polynomially reducible to the universality problem.

Proof. Upper boundedness of the series realised by an automaton $\mathcal{A}$ over $\mathbb{N}_{min}$ is equivalent to upper boundedness of the series realised by an automaton $\mathcal{B}$ over $\mathbb{N}_{min}$ with weights from the set $\{0, 1, \infty\}$ obtained by replacing all positive integer weights by 1. K. Hashiguchi showed that such automaton is upper bounded iff it is upper bounded by $2^{4n^3+n\log(n+2)+n}$ where n is the number of states of $\mathcal{B}$ [10]. Thus, automaton $\mathcal{B}$ is upper bounded iff the answer to the instance of the universality problem with a threshold $T = 2^{4n^3+n\log(n+2)+n}$ is affirmative. The input transformation can be done in polynomial time since T can be stored in $O(n^3)$ of space. $\square$

Chapter 2

Branchless Automata

One of the structurally restricted classes of weighted automata we examine is the class of acyclic automata with loops, which we study in the next chapter. Acyclic automata with loops are defined as automata which do not have a cycle longer than one.

Before that, we focus on a more restricted class of weighted automata – branchless automata – which is useful for characterising rational series realised by acyclic automata with loops and for defining a normal form of acyclic automata with loops, which we call branchless normal form.

In this chapter, we first define branchless automata and homogeneous branchless automata. Then, we characterise a subclass of rational series realised by branchless automata and define a normal form for branchless automata over commutative semirings. Next, we define notations and provide a necessary basis that we use in proving the main results of this thesis. These results include time complexity improvements for deciding certain decision problems, e.g., the upper boundedness problem or universality problem, for homogeneous branchless automata over semiring $\mathbb{N}_{min}$. In the last section, we show an improved version of the **NP**-hardness proof for the $\exists$-exact problem demonstrated in [2].

Some of the definitions, propositions and theorems we present are considered over arbitrary semiring or arbitrary commutative semiring, but most of them hold only for tropical semirings $\mathbb{N}_{min}$ and $\mathbb{Z}_{min}$.

A branchless automaton has one initial state, one terminal state, and at most one ingoing and outgoing transition for each state, except for loops.

Definition 2.0.1. Let S be a semiring and Σ an alphabet. A weighted automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ over S and Σ is called *branchless* if

(i) $\iota(1) \neq 0, \tau(n) \neq 0, \iota(2) = \ldots = \iota(n) = \tau(1) = \ldots = \tau(n-1) = 0,$

(ii) $\sigma(p, a, q) = 0$ holds for every $p, q \in [\![1, n]\!]$ with $q \notin \{p, p+1\}$,

(iii) for every $p \in [\![1, n-1]\!]$, there is at most one letter $c \in \Sigma$ such that $(p, c, p+1)$ is a transition.

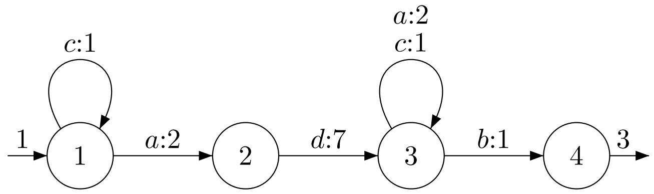

Figure 2.1: Branchless automaton $\mathcal{A}$ over semiring $(\mathbb{N}, +, \cdot, 0, 1)$ and alphabet $\{a, b, c, d\}$.

Definition 2.0.2. Let S be a semiring, Σ an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a branchless weighted automaton over S and Σ. A transition (p, c, q) where $p, q \in [\![1, n]\!]$ and $c \in \Sigma$ is called a *sequel* transition if $p \neq q$ and a *loop* transition if $p = q$.

We also define a more strict version of branchless automata – *homogeneous* branchless automata – which are simply branchless automata such that the labels of all sequel transitions are the same.

Definition 2.0.3. Let S be a semiring and Σ an alphabet. A branchless weighted automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ over S and Σ is called *homogeneous* if there exists a letter $a \in \Sigma$ such that for every $p, q \in [\![1, n]\!]$ and $c \in \Sigma$, $c = a$ holds for all transitions (p, c, q) such that $p \neq q$.

Convention 2.0.4. Without loss of generality, we denote the common label of all sequel transitions in a homogeneous branchless automaton by a. A loop with a label a is called an a-loop.

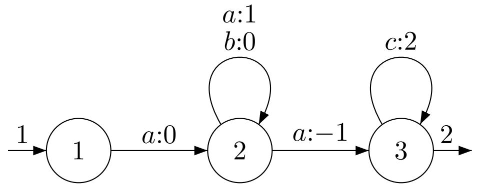

Figure 2.2: Homogeneous branchless automaton $\mathcal{B}$ over $\mathbb{Z}_{min}$ and $\{a, b, c\}$.

Definition 2.0.5. Let S be a semiring, Σ an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a branchless automaton over S and Σ. For every $p, q_1, q_2 \in [\![1, n]\!]$, the state p is said to be *between* the state q_1 and the state q_2 if $q_1 \leq p \leq q_2$ or $q_1 \geq p \geq q_2$.

2.1 Characterisation of Realised Series and Normal Form

Now, we characterise the subclass of rational series realised by branchless automata.

Theorem 2.1.1. Let S be a semiring and Σ an alphabet. For a rational series $r \in S\langle\langle\Sigma^*\rangle\rangle$, there exists a branchless weighted automaton $\mathcal{A}$ such that $||\mathcal{A}|| = r$ iff

$$r = v_0 R_1^* v_1 a_1 R_2^* v_2 \dots v_{n-1} a_{n-1} R_n^* v_n$$

for some $n \in \mathbb{N}^+, v_0, \dots, v_n \in S, a_1, \dots, a_{n-1} \in \Sigma$ and $R_1, \dots, R_n \in S\langle\Sigma\rangle$.[1]

Proof.

- $\Rightarrow$ For a given branchless automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ over S and Σ, we construct the series r as follows. Let $v_0 = \iota(1)$, $v_n = \tau(n)$, and for every $i \in [\![1, n-1]\!]$ and $c \in \Sigma$, such that $(i, c, i+1)$ is a sequel transition in $\mathcal{A}$, let $v_i = \sigma(i, c, i+1)$ and $a_i = c$. For every $i \in [\![1, n]\!]$, let $R_i \in S\langle\Sigma\rangle$ be given for every $c \in \Sigma$ by $(R_i, c) = \sigma(i, c, i)$. Obviously, $r = v_0 R_1^* v_1 a_1 R_2^* v_2 \dots v_{n-1} a_{n-1} R_n^* v_n = ||\mathcal{A}||$.
- $\Leftarrow$ For a given rational series $r = v_0 R_1^* v_1 a_1 R_2^* v_2 \dots v_{n-1} a_{n-1} R_n^* v_n$, we construct a branchless automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ over S and Σ such that $\iota(1) = v_0, \tau(n) = v_n$ and $\sigma(i, a_i, i+1) = v_i$ holds for all $i \in [\![1, n-1]\!]$. For every $i \in [\![1, n]\!]$ and every $c \in \Sigma$, let $\sigma(i, c, i) = (R_i, c)$. Every other value of σ, ι and τ equals to 0. It is obvious that $||\mathcal{A}|| = r$. $\square$

Example 2.1.2. The automaton $\mathcal{A}$ from Figure 2.1 realises the rational series

$$||\mathcal{A}|| = 1(1c)^* 2a7d(2a + 1c)^* 1b3$$

and since $(\mathbb{N}, +, \cdot, 0, 1)$ is a commutative semiring, we have

$$||\mathcal{A}|| = 42c^* ad(2a + c)^* b.$$

The automaton $\mathcal{B}$ from Figure 2.2 realises the rational series

$$||\mathcal{B}|| = 2a(1a + 0b)^* a(2c)^*.$$

Since every successful run of a branchless automaton over a commutative semiring must pass through all sequel transitions, we may "relocate" all weights of the sequel transitions, the initial weight and the terminal weight to the initial weight of the first state without changing the weight of any successful run of the given automaton. Based on that, we define the following normal form.

[1] Note that the support of a formal power series from $S\langle\Sigma\rangle$ is a subset of Σ.

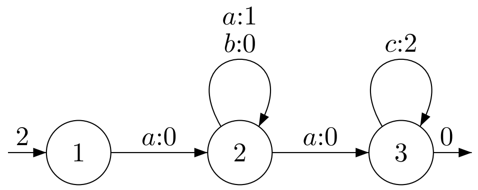

Figure 2.3: An automaton in normal form, equivalent to the automaton from Figure 2.2.

Definition 2.1.3. Let S be a commutative semiring and Σ an alphabet. A branchless automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ over S and Σ is in *normal form* if $\tau(n) = 1$ and $\sigma(p, c, q) = 1$ for every sequel transition (p, c, q).

Theorem 2.1.4. Let S be a commutative semiring, Σ an alphabet. For every branchless automaton $\mathcal{A} = (n, \sigma, \iota, \tau)$ over S and Σ, there exists a branchless automaton $\mathcal{A}' = (n, \sigma', \iota', \tau')$ over S and Σ in the normal form such that $||\mathcal{A}'|| = ||\mathcal{A}||$.

Proof. Let $c_1, \dots, c_{n-1}$ be the sequel transition letters of the automaton $\mathcal{A}$.[2] We put

$$\tau'(n) = 1$$

$$\iota'(1) = \iota(1)\tau(n)\prod_{i=1}^{n-1}\sigma(i, c_i, i+1)$$

$$\sigma'(i, c, i) = \sigma(i, c, i) \text{ for every } i \in [\![1, n]\!] \text{ and every } c \in \Sigma$$

$$\sigma'(i, c, j) = \begin{cases} 1 \text{ if } \sigma(i, c, j) \neq 0, \\ 0 \text{ otherwise,} \end{cases} \quad \text{for every } i, j \in [\![1, n]\!] \text{ and every } c \in \Sigma \text{ such that } i \neq j.$$

Every other value of ι' and τ' is set to zero.

For every successful run γ of $\mathcal{A}$ we construct exactly the same run γ' of $\mathcal{B}$ and show that $||\gamma'|| = ||\gamma||$. We see that

$$\begin{aligned} ||\gamma'|| &= \iota'(1)\sigma'(\gamma')\tau'(n) \\ &= \iota'(1)\prod_{i=1}^{n-1}\sigma'(i, c_i, i+1)L\tau'(n) \\ &\overset{(1)}{=} \iota'(1)L \\ &\overset{(2)}{=} \iota(1)\tau(n)\prod_{i=1}^{n-1}\sigma(i, c_i, i+1)L \\ &= \iota(1)\sigma(\gamma)\tau(n) \\ &= ||\gamma||, \end{aligned}$$

[2] Whenever we consider labels of the sequel transitions of an automaton, we assume they are sorted according to the order in which they occur in the automaton from the initial state to the terminal state.

where L is the weight of the loop transitions in γ or γ' which is the same for both runs. Here, (1) holds because $\tau'(n) = \prod_{i=1}^{n-1} \sigma'(i, c_i, i+1) = 1$. Equality (2) holds because $\iota'(1) = \iota(1)\tau(n)\prod_{i=1}^{n-1} \sigma(i, c_i, i+1)$.

Since we constructed a weight preserving bijection between the runs of $\mathcal{A}'$ and $\mathcal{A}$, we proved $||\mathcal{A}'|| = ||\mathcal{A}||$. □

Remark 2.1.5. Later on in this chapter, we assume every given branchless automaton over a commutative semiring to be in the normal form. This enables us to ignore all sequel transitions when resolving the weight of a run.

2.2 Definitions and Basic Observations

Before we prove the main theorems, we make some basic observations on branchless automata and define mappings useful in proving the main theorems. The section is divided according to mappings defined in the particular subsections.

2.2.1 Destinations of Runs

We define a mapping dest_γ for a given run γ, which returns a state reached after reading a given word along the run.

Definition 2.2.1. Let S be a semiring, Σ an alphabet, $\mathcal{A} = (Q, \sigma, \iota, \tau)$ a weighted automaton[3] over S and Σ and $\gamma = (q_1, a_1, q_2, \ldots, a_{m-1}, q_m)$ a run of $\mathcal{A}$ where $m \in \mathbb{N}^+$, $q_1, \ldots, q_m \in Q$, and $a_1, \ldots, a_{m-1} \in \Sigma$. The partial mapping $\mathsf{dest}_\gamma : \Sigma^* \to Q$ maps every prefix u of the word $w = a_1 \ldots a_{m-1}$ to the state reached after reading the prefix u along the run γ. Formally

$$\mathsf{dest}_\gamma(a_1 \ldots a_j) = q_{j+1}$$

for every $j \in [\![1, m-1]\!]$. The mapping dest_γ is defined only for prefixes of w.

Example 2.2.2. Let $\gamma = (1, c, 1, b, 2, a, 3, a, 4, a, 4, d, 4, a, 4, c, 5)$ be a run of the automaton $\mathcal{C}$ from Figure 2.4. Then

$$\begin{array}{lll} \mathsf{dest}_\gamma(\varepsilon) = 1, & \mathsf{dest}_\gamma(c) = 1, & \mathsf{dest}_\gamma(cb) = 2, \\ \mathsf{dest}_\gamma(cba) = 3, & \mathsf{dest}_\gamma(cbaa) = 4, & \mathsf{dest}_\gamma(cbaaa) = 4, \\ \mathsf{dest}_\gamma(cbaaad) = 4, & \mathsf{dest}_\gamma(cbaaada) = 4, & \mathsf{dest}_\gamma(cbaaadac) = 5. \end{array}$$

For every other word $w \in \Sigma^*$, $\mathsf{dest}_\gamma(w)$ is undefined.

[3] Notice that we do not assume branchless automata.

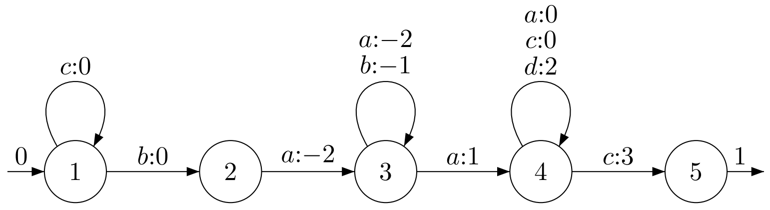

Figure 2.4: Branchless automaton $\mathcal{C}$ over $\mathbb{Z}_{min}$ and $\Sigma = \{a, b, c, d\}$.

2.2.2 Number of Loop Transitions for a Given State and Letter

Next, we define a mapping num_γ that is useful in resolving the weight of a run and detecting the number of uses of a particular loop.

Definition 2.2.3. Let S be a semiring, Σ an alphabet, $\mathcal{A} = (Q, \sigma, \iota, \tau)$ a branchless automaton over S and Σ and $\gamma = (q_1, a_1, \ldots, a_{m-1}, q_m)$ a run of $\mathcal{A}$ where $m \in \mathbb{N}^+$, $a_1, \ldots, a_{m-1} \in \Sigma$ and $q_1, \ldots, q_m \in Q$. The mapping $\mathsf{num}_\gamma : Q \times \Sigma \to \mathbb{N}$ maps a state $q \in Q$ and a letter $c \in \Sigma$ to the number of times the c-labelled loop is used by the run γ in the state q. Formally

$$\mathsf{num}_\gamma(q, c) = |\{i \in [\![1, m-1]\!] \mid q_i = q = q_{i+1} \text{ and } a_i = c\}|.$$

Example 2.2.4. Let $\gamma = (1, c, 1, b, 2, a, 3, a, 4, a, 4, d, 4, a, 4, c, 5)$ be a run of the automaton $\mathcal{C}$ from Figure 2.4. Then, for instance,

$$\begin{array}{lll} \mathsf{num}_\gamma(1, c) = 1, & \mathsf{num}_\gamma(1, b) = 0, & \mathsf{num}_\gamma(2, a) = 0, \\ \mathsf{num}_\gamma(3, a) = 0, & \mathsf{num}_\gamma(4, a) = 2, & \mathsf{num}_\gamma(4, b) = 0, \\ \mathsf{num}_\gamma(4, c) = 0, & \mathsf{num}_\gamma(4, d) = 1, & \mathsf{num}_\gamma(5, b) = 0. \end{array}$$

Proposition 2.2.5. Let S be a commutative semiring, Σ an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a branchless automaton over S and Σ. For every word $w \in \Sigma^*$ and every successful run γ of $\mathcal{A}$ on w, $\sigma(\gamma) = \prod_{p=1}^{n} \prod_{c \in \Sigma} \sigma(p, c, p)^{\mathsf{num}_\gamma(p,c)}$. In the case of tropical semirings $\mathbb{N}_{min}$ and $\mathbb{Z}_{min}$,

$$\sigma(\gamma) = \sum_{p=1}^{n} \sum_{c \in \Sigma} \mathsf{num}_\gamma(p, c)\sigma(p, c, p).$$

Proof. Since we assume all given branchless automata to be in the normal form, then all sequel transitions have a weight of 1.[4] Therefore the only relevant weights are on the loops. $\square$

Corollary 2.2.6. Let S be a commutative semiring, Σ an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a branchless automaton over S and Σ such that $\sigma(p, c, p) \in \{0, 1\}$ for every $p \in$

[4]The number 1 denotes the neutral element of a multiplicative operation.

$[\![1, n]\!], c \in \Sigma - \{a\}$. Then, for every word $w \in \Sigma^*$ and every successful run γ on w of $\mathcal{A}$, $\sigma(\gamma) = \prod_{p=1}^{n} \sigma(p, a, p)^{\mathsf{num}_\gamma(p,a)}$. In the case of tropical semirings $\mathbb{N}_{min}$ or $\mathbb{Z}_{min}$,

$$\sigma(\gamma) = \sum_{p=1}^{n} \mathsf{num}_\gamma(p, a)\sigma(p, a, p).$$

2.2.3 Loop Sets at the Particular States

The following mapping loops returns for a given state of a *homogeneous* branchless automaton the set of letters for which there exists a loop on a given state except for the letter a. This mapping is used for example in constructing a run (e.g., Lemma 2.3.12 or Lemma 2.4.3) or in defining a relation of supersession.

Definition 2.2.7. Let S be a semiring, Σ an alphabet and $\mathcal{A} = (Q, \sigma, \iota, \tau)$ a homogeneous branchless automaton over S and Σ. The mapping $\mathsf{loops} : Q \to 2^\Sigma$ maps a state $p \in Q$ to the set of letters other than a, for which there exists a loop transition at p. Formally,

$$\mathsf{loops}(q) = \{c \in \Sigma - \{a\} \mid \sigma(q, c, q) \neq 0\}$$

for every state $q \in Q$.

Example 2.2.8. For the automaton $\mathcal{C}$ from Figure 2.4, the following holds:

$$\mathsf{loops}(1) = \{c\}, \quad \mathsf{loops}(2) = \emptyset, \quad \mathsf{loops}(3) = \{b\}, \quad \mathsf{loops}(4) = \{c, d\}, \quad \mathsf{loops}(5) = \emptyset.$$

2.2.4 Supersession

We now define a notion of *supersession*, which is a tool heavily used in our proof that the upper boundedness problem for homogeneous branchless automata over $\mathbb{N}_{min}$ belongs to **P** (see Theorem 2.3.16).

Definition 2.2.9. Let Σ be an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over $\mathbb{N}_{min}$ and Σ. A state $p \in [\![1, n]\!]$ with an a-loop is *superseded* by a state $q \in [\![1, n]\!]$ with an a-loop, if for every state $r \in [\![1, n]\!]$ between p and q, $\mathsf{loops}(p) \subseteq \mathsf{loops}(r) \subseteq \mathsf{loops}(q)$.[5] We also say that q supersedes p and write $q \succeq p$. If $\sigma(q, a, q) = 0$, then we say q supersedes p with a zero loop and write $q \succeq^0 p$, otherwise we say q supersedes p with a positive loop and write $q \succeq^+ p$.

The relation $\succeq$ introduced above is a preorder relation on the set of states with an a-loop. The reflexivity is obvious. The transitivity is proved in the following proposition.

[5] Note that ∞ is the neutral element of the additive operation of $\mathbb{N}_{min}$.

Proposition 2.2.10. Let Σ be an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over $\mathbb{N}_{min}$ and Σ. For every $p, q, r \in [\![1, n]\!]$ such that there is an a-loop on p, q and r, $p \succeq r$ holds whenever both $p \succeq q$ and $q \succeq r$.

Proof. Let $p \succeq q$ and $q \succeq r$. We need to show that for every state t between p and r, $\mathsf{loops}(p) \overset{(A)}{\supseteq} \mathsf{loops}(t) \overset{(B)}{\supseteq} \mathsf{loops}(r)$. If $p \leq q \leq r$ or $r \leq q \leq p$, the state t must be between p and q or between q and r. If $p \leq r \leq q$ or $q \leq r \leq p$, the state t must be between p and q. Finally, if $q \leq p \leq r$ or $r \leq p \leq q$, the state t must be between q and r. It is, therefore, sufficient to check that (A) and (B) holds for every state between p and q and also for every state between q and r.

(A) For every state t between p and q, $\mathsf{loops}(p) \supseteq \mathsf{loops}(t)$ holds by $p \succeq q$. For every state t between q and r, $\mathsf{loops}(p) \overset{(1)}{\supseteq} \mathsf{loops}(q) \overset{(2)}{\supseteq} \mathsf{loops}(t)$. Here, (1) follows from $p \succeq q$ and (2) from $q \succeq r$.

(B) For every state t between p and q, $\mathsf{loops}(t) \overset{(1)}{\supseteq} \mathsf{loops}(q) \overset{(2)}{\supseteq} \mathsf{loops}(r)$. Here, (1) follows from $p \succeq q$ and (2) from $q \succeq r$. For every state t between q and r, $\mathsf{loops}(t) \supseteq \mathsf{loops}(r)$ holds by $q \succeq r$. $\square$

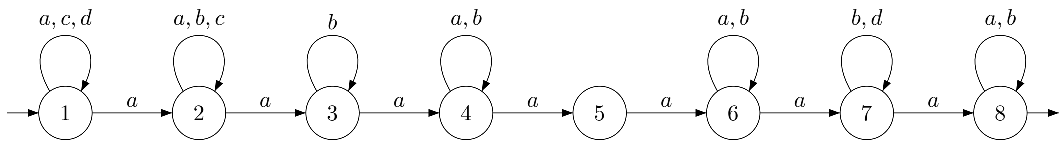

Figure 2.5: A branchless automaton $\mathcal{D}$ with hidden weights over alphabet $\{a, b, c, d\}$.

Example 2.2.11. For automaton $\mathcal{D}$ from Figure 2.5, the states which are considered in the supersession relations are $1, 2, 4, 6, 8$, because they have a loop on the letter a. The only supersession relation here is $2 \succeq 4$. For other pairs of states, some condition is not satisfied. For example, state 1 does not supersede any state, as $\mathsf{loops}(1) \not\supseteq \mathsf{loops}(2)$. States 4 and 6 are not in a relation since $\mathsf{loops}(5) \not\supseteq \mathsf{loops}(4)$ nor $\mathsf{loops}(5) \not\supseteq \mathsf{loops}(6)$. Also, states 6 and 8 are not in a relation, because $\mathsf{loops}(6) \not\supseteq \mathsf{loops}(7)$ and $\mathsf{loops}(8) \not\supseteq \mathsf{loops}(7)$.

2.2.5 Minimal Runs

In branchless automata, after reading a letter $c \in \Sigma$ in a state p, a run cannot proceed to a state different than p or $p + 1$. In the case of homogeneous branchless automata, a run must stay in the same state after reading a letter $c \in \Sigma - \{a\}$.

Proposition 2.2.12. Let S be a semiring, Σ an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over S and Σ. For every word $w \in \Sigma^*$ and states $p, q \in [\![1, n]\!]$ such that $p \xrightarrow{w} q$, the inequalities $p \leq q \leq p + \#_a(w)$ hold.

Proof. In a homogeneous branchless automaton, the run may "move" at most $\#_a(w)$ states further while reading $\#_a(w)$ times the letter a. □

Proposition 2.2.13. Let S be a semiring, Σ an alphabet, $\mathcal{A} = (n, \sigma, \iota, \tau)$ a branchless automaton over S and Σ, and $c_1, \ldots, c_{n-1} \in \Sigma$ the labels of the sequel transitions of $\mathcal{A}$. For every word $w \in \Sigma^*$ and states $p, q \in [\![1, n]\!]$ such that $p \xrightarrow{w} q$, the w must contain a scattered subword $c_p \ldots c_{q-1}$.

Proof. The proof is trivial just by looking at a rational series realised by $\mathcal{A}$. □

Definition 2.2.14. Let S be a tropical semiring $\mathbb{N}_{min}$ or $\mathbb{Z}_{min}$, Σ an alphabet, $\mathcal{A} = (n, \sigma, \iota, \tau)$ a branchless automaton over S and Σ, $p, q \in [\![1, n]\!]$ states and $w \in \Sigma^*$ a word. A run γ on w from p to q is called *minimal* if $\sigma(\gamma) \leq \sigma(\gamma')$ holds for every run γ' on w from p to q.

Definition 2.2.15. Let S be a tropical semiring $\mathbb{N}_{min}$ or $\mathbb{Z}_{min}$, Σ an alphabet, $\mathcal{A} = (n, \sigma, \iota, \tau)$ a branchless automaton over S and Σ, $\gamma = (q_1, a_1, \ldots, a_{m-1}, q_m)$ a run of $\mathcal{A}$ where $m \in \mathbb{N}^+$, $a_1, \ldots, a_{m-1} \in \Sigma$, and $q_1, \ldots, q_m \in [\![1, n]\!]$. A run γ' of $\mathcal{A}$ is called a *subrun* of γ in $\mathcal{A}$ if there exists $k, l \in [\![1, m]\!]$ such that $k \leq l$ and $\gamma' = (q_k, a_k, \ldots, a_{l-1}, q_l)$.

Proposition 2.2.16. Let S be a tropical semiring $\mathbb{N}_{min}$ or $\mathbb{Z}_{min}$, Σ an alphabet, $\mathcal{A} = (n, \sigma, \iota, \tau)$ a branchless automaton over S and Σ, $p, q \in [\![1, n]\!]$ states of $\mathcal{A}$ and $w \in \Sigma^*$ a word. If γ is a minimal run from p to q on w and $\gamma_{r,s,v}$ is a subrun of γ from some state $r \in [\![1, n]\!]$ to some state $s \in [\![1, n]\!]$ on a word $v \in \Sigma^*$, then $\gamma_{r,s,v}$ is minimal run as well.

Proof. We prove the proposition by a contradiction. Let γ be a run from p to q on w and let $\gamma_{r,s,v}$ be a subrun of γ from p to q on v for some $r, s, \in [\![1, n]\!]$ and $v \in \Sigma^*$. Let $u_1, u_2 \in \Sigma^*$ be words such that $w = u_1 v u_2$, γ_{p,r,u_1} a subrun of γ from p to r on u_1 and γ_{s,q,u_2} a subrun of γ from s to q on u_2. If the run $\gamma_{r,s,v}$ is not minimal, then there exists a run $\gamma'_{r,s,v}$ for which $\sigma(\gamma'_{r,s,v}) < \sigma(\gamma_{r,s,v})$. Then, we can construct a run γ' from p to q on w such that γ' uses the run γ_{p,r,u_1} to get from p to r on u_1, the run $\gamma'_{r,s,v}$ to get from r to s on v and the run γ_{s,q,u_2} to get from s to q on u_2. Then

$$\sigma(\gamma') = \sigma(\gamma_{p,r}) + \sigma(\gamma'_{r,s}) + \sigma(\gamma_{s,q}) < \sigma(\gamma_{p,r}) + \sigma(\gamma_{r,s}) + \sigma(\gamma_{s,q}) = \sigma(\gamma)$$

which is a contradiction with γ being a minimal run on w from p to q. □

2.2.6 Representative Words

Below defined *representative words* try to act as the "hardest possible word", which can be read in a given state. They are used mainly in proving the main theorem about the universality problem and some constructions (e.g., Lemma 2.3.4).

Definition 2.2.17. Let S be a semiring, Σ an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over S and Σ. For every state $i \in [\![2, n]\!]$ with $\mathsf{loops}(i) = \{c_1, \ldots, c_j\}$ where $j \in \mathbb{N}$ and letters $c_1, \ldots, c_j$ are listed in according to some fixed ordering on Σ, *the representative word of the state* i, denoted as ω_i, is defined by $\omega_i = ac_1 \ldots c_j$. For the state 1, if $(1, a, 1)$ is not a transition then $\omega_1 = \varepsilon$, otherwise $\omega_1 = ac_1 ... c_j$ with $\mathsf{loops}(1) = \{c_1, \ldots, c_j\}$ where $j \in \mathbb{N}$ and letters $c_1, \ldots, c_j$ are listed in according to mentioned ordering. The language $\omega_1^* \omega_2^* \ldots \omega_n^*$ is called *the representative language of the automaton* $\mathcal{A}$ denoted by $\Omega_{\mathcal{A}}$.

Definition 2.2.18. Let S be a semiring, Σ an alphabet, $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over S and Σ, γ a run of $\mathcal{A}$ and $i, q \in [\![1, n]\!]$. We say that a word ω_i is *read in the state* q in a run γ if after reading the first letter of ω_i (which is a), γ gets to the state q.

Example 2.2.19. We consider the automaton $\mathcal{D}$ from Figure 2.5 and we sort alphabet standardly. Then

$$\omega_1 = acd, \quad \omega_2 = abc, \quad \omega_3 = ab, \quad \omega_4 = ab, \quad \omega_5 = a, \quad \omega_6 = ab, \quad \omega_7 = abd, \quad \omega_8 = ab.$$

Proposition 2.2.20. Let S be a semiring, Σ an alphabet, $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over S and Σ. If $p \xrightarrow{\omega_i} p$ (or $p-1 \xrightarrow{\omega_i} p$) for some $p, i \in [\![1, n]\!]$ ($p \in [\![2, n]\!]$, $i \in [\![1, n]\!]$) then $\mathsf{loops}(p) \supseteq \mathsf{loops}(i)$.

Proof. Since $\mathsf{alph}(\omega_i) = \mathsf{loops}(i) \cup \{a\}$ and a is the first letter of the word ω_i, the state p has to have loops upon all letters from $\mathsf{alph}(\omega_i) - \{a\}$, meaning that $\mathsf{loops}(p) \supseteq \mathsf{alph}(\omega_i) - \{a\} = \mathsf{loops}(i)$. $\square$

2.3 The Upper Boundedness Problem

In this section, we prove multiple lemmas that finally lead us to prove that the problem of deciding upper boundedness of series realised by homogeneous branchless automata over $\mathbb{N}_{min}$ belongs to **P**.

Lemma 2.3.1. Let S be a tropical semiring $\mathbb{N}_{min}$ or $\mathbb{Z}_{min}$, Σ be an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a branchless automaton over S and Σ. If there exists $i \in [\![1, n]\!]$ and

$c \in \Sigma$ such that (i, c, i) is a transition with a positive weight and there does not exist a state $j \in [\![1, n]\!] - \{i\}$ such that (i, c, j) or (j, c, i) is a transition, then the series realised by the automaton $\mathcal{A}$ is not upper bounded.

Proof. Let $c_1, \ldots, c_{n-1}$ be the labels of the sequel transitions of the automaton $\mathcal{A}$. If $1 < i < n$ then $c_{i-1} \neq c, c \neq c_i$, as otherwise $(i-1, c, i)$ or $(i, c, i+1)$ would be a transition, contradicting the assumptions. Similarly, if $i = 1$ then $c_1 \neq c$ and if $i = n$, then $c_{n-1} \neq c$.

For any $T \in \mathbb{N}$, we take a word

$$w = c_1 \ldots c_{i-1} c^T c_i \ldots c_{n-1}$$

and we show that $(||\mathcal{A}||, w) \geq T$, which proves the lemma. For now, we assume $1 < i < n$.

Let γ be a successful run on w. For a contradiction, assume that $\mathsf{dest}_\gamma(c_1 \ldots c_{i-1}) < i$. Then, by Proposition 2.2.13, the rest of the word w (i.e., $c^T c_i \ldots c_{n-1}$) must contain $c_{i-1} \ldots c_{n-1}$ as a scattered subword. But $\#_{c_{i-1}}(c^T c_i \ldots c_{n-1}) < \#_{c_{i-1}}(c_{i-1} c_i \ldots c_{n-1})$ since $c \neq c_{i-1}$ and therefore $\mathsf{dest}_\gamma(c_1 \ldots c_{i-1}) \geq i$.

By Proposition 2.2.13, we also have that if $1 \xrightarrow{c_1 \ldots c_{i-1} c^T} q$ for $q > i$, then $c_1 \ldots c_{i-1} c^T$ must contain $c_1 \ldots c_{q-1}$ as a scattered subword. But $\#_{c_i}(c_1 \ldots c_{i-1} c^T) < \#_{c_i}(c_1 \ldots c_{q-1})$ since $q > i$ and $c_i \neq c$. Therefore $\mathsf{dest}_\gamma(c_1 \ldots c_{i-1}) \leq i$.

We showed that for every successful run γ on w, $\mathsf{dest}_\gamma(c_1 \ldots c_{i-1}) = i = \mathsf{dest}_\gamma(c_1 \ldots c_{i-1} c^T)$. Since $\sigma(i, c, i) > 0$, the weight of the word c^T in γ is greater than T which proves that $(||\mathcal{A}||, w) > T$.

For $i = 1$ or $i = n$, we only need to prove that $\mathsf{dest}_\gamma(c_1 \ldots c_{i-1}) \leq i$ or $\mathsf{dest}_\gamma(c_1 \ldots c_{i-1}) \geq i$ respectively, which is done in the same way as above. $\square$

Corollary 2.3.2. Let S be a tropical semiring $\mathbb{N}_{min}$ or $\mathbb{Z}_{min}$, Σ an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over S and Σ. If there exists $p \in [\![1, n]\!]$ and $c \in \Sigma - \{a\}$ such that (p, c, p) is a transition with positive weight, then the series realised by the automaton $\mathcal{A}$ is not upper bounded.

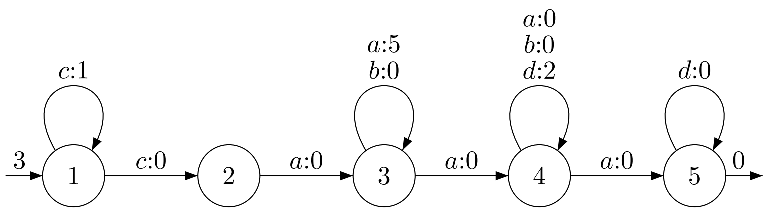

Figure 2.6: Branchless automaton $\mathcal{A}$ over $\mathbb{N}_{min}$ and alphabet $\{a, b, c, d\}$.

Example 2.3.3. Let $\mathcal{A}$ be the automaton from Figure 2.6. We see that state 4 has a positive loop on the letter d and both transitions from or to state 4 do not have label d. Therefore we can construct a set of coefficients $(||\mathcal{A}||, aaad^*a)$ which is not upper bounded and thus prove the unboundedness of the series realised by $\mathcal{A}$. Notice that although state 1 does not satisfy the conditions of Lemma 2.3.1, we can still construct a set of coefficients $(||\mathcal{A}||, c^*caaa)$ which is not upper bounded. This observation hints to us that deciding upper boundedness consists of something more complex, which takes into account other states.

Lemma 2.3.4. Let S be a tropical semiring $\mathbb{N}_{min}$ or $\mathbb{Z}_{min}$, Σ an alphabet, $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over S and Σ, $i \in [\![2, n]\!]$ a state and γ a run of $\mathcal{A}$ on $\omega_2 \dots \omega_i$ such that $\mathsf{dest}_\gamma(\omega_2 \dots \omega_i) = j < i$. Let $k \in [\![1, i-1]\!]$ be the smallest state such that $\mathsf{dest}_\gamma(\omega_2 \dots \omega_{k+1}) < k + 1$. Then, state k has an a-loop and $\mathsf{loops}(k) \supseteq \mathsf{loops}(l)$ holds for every $l \in [\![k, i]\!]$.

Proof. Let us assume that $k > 1$. Since k is the smallest state with the mentioned property, we have $\mathsf{dest}_\gamma(\omega_2 \dots \omega_k) \geq k$. Then, using Proposition 2.2.12 with the word ω_{k+1} and states $\mathsf{dest}_\gamma(\omega_2 \dots \omega_k)$ and $\mathsf{dest}_\gamma(\omega_2 \dots \omega_{k+1})$, we have

$$\mathsf{dest}_\gamma(\omega_2 \dots \omega_k) = k = \mathsf{dest}_\gamma(\omega_2 \dots \omega_{k+1}). \tag{2.1}$$

By (2.1) and by $\#_a(\omega_{k+1}) = 1$, we see that the state k has an a-loop. In the case of $k = 1$, we have $\mathsf{dest}_\gamma(\omega_2) = 1$ which is sufficient to see that state 1 has an a-loop.

For every $l \in [\![k+1, i]\!]$, if $m \in [\![1, n]\!]$ is a state such that $m = \mathsf{dest}_\gamma(\omega_2 \dots \omega_l)$, then $\mathsf{loops}(m) \supseteq \mathsf{loops}(l)$ by Proposition 2.2.20. Also by (2.1) and by Proposition 2.2.12 with word $\omega_{k+2} \dots \omega_l$ and states $\mathsf{dest}_\gamma(\omega_2 \dots \omega_{k+1})$ and $\mathsf{dest}_\gamma(\omega_2 \dots \omega_l)$, we have that

$$\begin{aligned} &k \leq m \leq k + \#_a(\omega_{k+2} \dots \omega_l) = l - 1 \\ &k \leq m < l. \end{aligned}$$

Therefore, for every state $l \in [\![k+1, i]\!]$ we can find a sequence of states $m_0, \dots, m_t \in [\![k, l]\!]$ for some $t \in \mathbb{N}^+$ such that

$$k = m_0 < m_1 < \dots < m_t = l$$
$$\mathsf{loops}(k) = \mathsf{loops}(m_0) \supseteq \mathsf{loops}(m_1) \supseteq \dots \supseteq \mathsf{loops}(m_t) = \mathsf{loops}(l)$$

which proves the second part of our lemma (when $l = k$, $\mathsf{loops}(k) \supseteq \mathsf{loops}(l)$ holds trivially). $\square$

Example 2.3.5. Let us consider the automaton $\mathcal{B}$ from Figure 2.7. We see that every run of $\mathcal{B}$ on the word $\omega_2\omega_3\omega_4$ from state 1 must end in state 4 because there does not

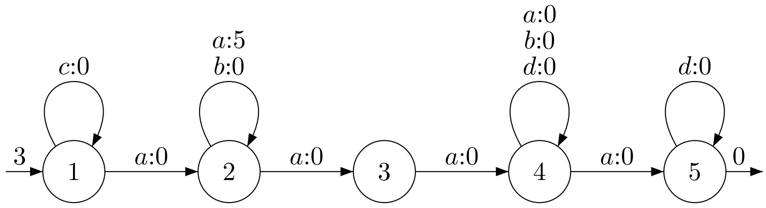

Figure 2.7: Branchless automaton $\mathcal{B}$ over $\mathbb{N}_{min}$ and $\{a, b, c, d\}$.

exist a state $p \in \{1, 2, 3\}$ such that $\mathsf{loops}(p) \supseteq \mathsf{loops}(q)$ for every $q \in [\![p, 4]\!]$. On the other hand, we may construct a run on the word $\omega_2\omega_3\omega_4\omega_5$ from state 1 to state 4 such that the word ω_5 is read in state 4. We clearly see that $\mathsf{loops}(4) \supseteq \mathsf{loops}(5)$.

The following lemma is a "reversed version" of Lemma 2.3.4. The only main difference is that the last inclusion does not hold for $l = i$.

Lemma 2.3.6. Let S be a tropical semiring $\mathbb{N}_{min}$ or $\mathbb{Z}_{min}$, Σ an alphabet, $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over S and Σ, $i \in [\![1, n-1]\!]$ a state and γ a run of $\mathcal{A}$ on $\omega_{i+1} \dots \omega_n$ from some state $j \in [\![i+1, n]\!]$ to n. Let $k \in [\![j, n]\!]$ be the biggest state such that $\mathsf{dest}_\gamma(\omega_{i+1} \dots \omega_{k-1}) > k - 1$. Then, state k has an a-loop and $\mathsf{loops}(k) \supseteq \mathsf{loops}(l)$ holds for every $l \in [\![i+1, k]\!]$.

Proof. Analogously as in Lemma 2.3.4, we can prove that

$$\mathsf{dest}_\gamma(\omega_2 \dots \omega_{k-1}) = k = \mathsf{dest}_\gamma(\omega_2 \dots \omega_k).$$

and that state k has an a-loop.

For every $l \in [\![i+1, k-1]\!]$, if $m \in [\![1, n]\!]$ is a state such that $m = \mathsf{dest}_\gamma(\omega_{i+1} \dots \omega_l)$, then $\mathsf{loops}(m) \supseteq \mathsf{loops}(l)$ by Proposition 2.2.20. Also by Proposition 2.2.12 with word $\omega_{l+1} \dots \omega_{k-1}$ and states $\mathsf{dest}_\gamma(\omega_2 \dots \omega_l)$ and $\mathsf{dest}_\gamma(\omega_2 \dots \omega_{k-1})$, we have that

$$\begin{aligned} m \le k \le m + \#_a(\omega_{l+1} \dots \omega_{k-1}) &= m + k - l - 1 \\ 0 &\le m - l - 1 \\ l < m &\le k. \end{aligned}$$

Therefore, for every state $l \in [\![i+1, k-1]\!]$ we can find a sequence of states $m_0, \dots, m_t \in [\![l, k]\!]$ for some $t \in \mathbb{N}^+$ such that

$$l = m_0 < m_1 < \dots < m_t = k$$
$$\mathsf{loops}(l) = \mathsf{loops}(m_0) \subseteq \mathsf{loops}(m_1) \subseteq \dots \subseteq \mathsf{loops}(m_t) = \mathsf{loops}(k)$$

which proves the second part of our lemma (when $l = k$, $\mathsf{loops}(k) \supseteq \mathsf{loops}(l)$ holds trivially). $\square$

Example 2.3.7. Let us consider the automaton $\mathcal{B}$ from Figure 2.7. We see that there exists a run of $\mathcal{B}$ on the word $\omega_2\omega_3\omega_4\omega_5$ from state 4 to 5 which reads the word $\omega_2\omega_3\omega_4$ in state 4 and that $\mathsf{loops}(4) \supseteq \mathsf{loops}(p)$ for every $p \in \{2, 3, 4\}$.

Lemma 2.3.8. Let Σ be an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over $\mathbb{N}_{min}$ and Σ. If there exists a state $p \in [\![1, n]\!]$ with an a-loop and there does not exist a state $q \in [\![1, n]\!]$ with an a-loop and $q \succeq^0 p$, then we can find a state $r \in [\![1, n]\!]$ such that (i) r has a positive a-loop, (ii) there is no state $s \in [\![1, n]\!]$ with an a-loop such that $s \succeq^0 r$ and (iii) for every state $s \in [\![1, n]\!]$ with an a-loop, if $s \succeq^+ r$, then $\mathsf{loops}(r) = \mathsf{loops}(t) = \mathsf{loops}(s)$ and $\sigma(t, a, t) \neq 0$ holds for every state t between r and s.

Proof. Let $M \subseteq [\![1, n]\!]$ be a set of states such that $q \in M$ iff q has a positive a-loop and there does not exist a state $s \in [\![1, n]\!]$ such that $s \succeq^0 q$. Let $r \in [\![1, n]\!]$ be the smallest state from M such that r is maximal in M in according to the inclusion of the values of the mapping loops. Now, we prove that r satisfies all conditions of Lemma.

(i) and (ii) are trivial, since $r \in M$.

(iii) Assume we have a state $s \in [\![1, n]\!]$ with an a-loop such that $s \succeq^+ r$ and $\mathsf{loops}(s) \supsetneq \mathsf{loops}(r)$. If $s \in M$, then r is not maximal according to the inclusion of mapping loops which is a contradiction. Therefore $s \notin M$. Since s has a positive a-loop by $s \succeq^+ r$, then the only possibility of $s \notin M$ is that there exists a state t with an a-loop such that $t \succeq^0 s$. But then, by transitivity of $\succeq$, we would get $t \succeq^0 r$ which is a contradiction with $r \in M$.

Therefore if $s \succeq^+ r$, then $\mathsf{loops}(s) = \mathsf{loops}(r)$, which together with the properties of $s \succeq^+ r$ implies that $\mathsf{loops}(r) = \mathsf{loops}(t) = \mathsf{loops}(s)$ for every state t between s and r. Then also $\sigma(t, a, t) \neq 0$, because otherwise, we would have $t \succeq^0 r$ which would be a contradiction. $\square$

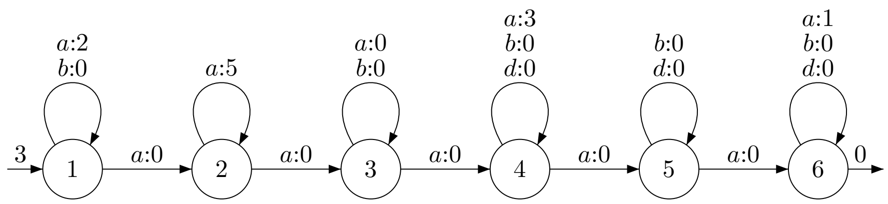

Figure 2.8: Branchless automaton $\mathcal{C}$ over $\mathbb{N}_{min}$ and $\{a, b, c, d\}$.

Example 2.3.9. For the automaton $\mathcal{C}$ of Figure 2.8, if we define M as in the proof of Lemma 2.3.8, then $M = \{1, 4, 6\}$. We can clearly see that states 4 and 6 are maximal according to the inclusion of the values of mapping loops and both states satisfy all three conditions of Lemma 2.3.8.

Lemma 2.3.10. Let Σ be an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over $\mathbb{N}_{min}$ and Σ. If there exists a state $i \in [\![1, n]\!]$ with a positive a-loop and there does not exist a state $j \in [\![1, n]\!]$ with an a-loop such that $j \succeq^0 i$, then the series realised by the automaton $\mathcal{A}$ is not upper bounded.

Proof. Let $p \in [\![1, n]\!]$ be the state whose existence is implied by Lemma 2.3.8.

For every $T \in \mathbb{N}$, let us take $w = xyz$, where

$$x = \omega_2 \dots \omega_p, \quad y = \omega_p^{T+n}, \quad z = \omega_{p+1} \dots \omega_n.$$

We show that $(||\mathcal{A}||, w) \geq T$, which proves the unboundedness. For every successful run γ on w, we have $\mathsf{dest}_\gamma(x) \leq p$ and $\mathsf{dest}_\gamma(xy) \geq p$ by Proposition 2.2.12. Let γ be a successful run on w and $q, r \in [\![1, n]\!]$ states such that $\mathsf{dest}_\gamma(x) = q$ and $\mathsf{dest}_\gamma(xy) = r$. If $q = p = r$ then $\sigma(\gamma) \geq T + n$, because y is read on state p which has a positive a-loop.

Let us assume $q < p$. Let $s \in [\![1, q]\!]$ be the state whose existence is implied by Lemma 2.3.4. By Lemma 2.3.4 and by (i) of Lemma 2.3.8, we know that s and p have an a-loop. Now, we want to show that $s \succeq p$. By Lemma 2.3.4, we know that $\mathsf{loops}(s) \supseteq \mathsf{loops}(j)$ holds for every $j \in [\![s, p]\!]$. For every $j \in [\![q, p]\!]$

$$\mathsf{loops}(j) \supseteq \mathsf{loops}(p) \tag{2.2}$$

holds since the word ω_p is read on every state $j \in [\![q, p]\!]$ and by Proposition 2.2.20.

Similarly as in Lemma 2.3.4, we can prove that for every $m \in [\![s + 1, q]\!]$, if $l \in [\![1, n]\!]$ is the smallest state such that $m = \mathsf{dest}_\gamma(\omega_2 \dots \omega_l)$, then $\mathsf{loops}(m) \supseteq \mathsf{loops}(l)$ and

$$m < l \leq p.$$

The second inequality follows from $\mathsf{dest}_\gamma(\omega_2 \dots \omega_p) = q$ which implies that for every $i \in [\![2, q]\!]$, there exists some $j \in [\![2, p]\!]$ such that $\mathsf{dest}_\gamma(\omega_2 \dots \omega_j) = i$.

Thus, for every state $m \in [\![s + 1, q]\!]$, we can find a sequence of states $l_0, \dots, l_k \in [\![m, p]\!]$ for some $k \in \mathbb{N}^+$ such that

$$m = l_0 < l_1 < \dots < l_t \geq q$$
$$\mathsf{loops}(m) = \mathsf{loops}(l_0) \supseteq \mathsf{loops}(l_1) \supseteq \dots \supseteq \mathsf{loops}(l_t).$$

By (2.2), we know that $\mathsf{loops}(l_t) \supseteq \mathsf{loops}(p)$. Thus we showed that $\mathsf{loops}(m) \supseteq \mathsf{loops}(p)$ for every state $m \in [\![s, q]\!]$, which altogether implies that $s \succeq p$.

By Lemma 2.3.8, we have that $s \succeq^+ p$, and $\mathsf{loops}(s) = \mathsf{loops}(j) = \mathsf{loops}(p)$ and $\sigma(j, a, j) > 0$ holds for every state $j \in [\![s, p]\!]$.

Analogously, using Lemma 2.3.6, we can prove that if $r > p$, then there exists a state $t \geq r$, such that $\mathsf{loops}(p) = \mathsf{loops}(j) = \mathsf{loops}(t)$ and $\sigma(j, a, j) > 0$ holds for every state $j \in [\![p, t]\!]$.[6] The established property is illustrated in Figure 2.9.

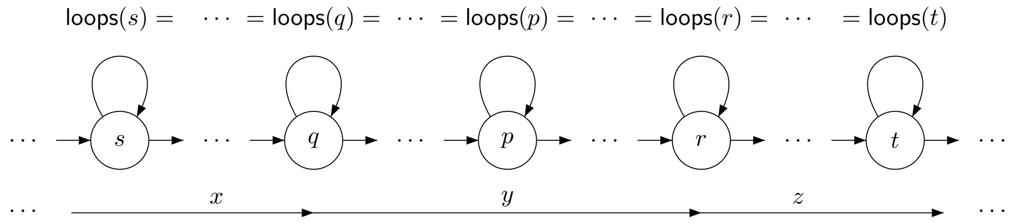

Figure 2.9: States of the automaton $\mathcal{A}$ from the proof of Lemma 2.3.10 with their loops values. The arrows denote that in the run γ, reading the word x ends in q, the word y is read from q to r, and reading the word z starts in r.

But then, the only possible way to read y between s and t is by using at least T positive a-loops, because the letter a may be read no more than n times using a sequel transition. Hence we have $(||\mathcal{A}||, w) \geq T$. □

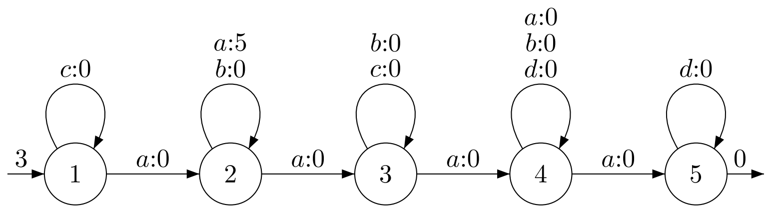

Figure 2.10: Branchless automaton $\mathcal{D}$ over $\mathbb{N}_{min}$ and alphabet $\{a, b, c, d\}$, where state 2 is not superseded by any other state.

Example 2.3.11. In automaton $\mathcal{D}$ from Figure 2.10, state 2 has a positive a-loop and it is not superseded by any other state. Therefore, for any $k \in \mathbb{N}$ we can construct words $x = \omega_2 = ab$, $y = \omega_2^{k+5} = (ab)^{k+5}$, $z = \omega_3\omega_4\omega_5 = abcabcdad$ for which $(||\mathcal{A}||, xyz) > k$ holds.

Lemma 2.3.12. Let Σ be an alphabet, $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over $\mathbb{N}_{min}$ and Σ, which has positive loops only on the letter a, $i \in [\![1, n]\!]$ a state with a positive a-loop, $j \in [\![1, n]\!]$ a state such that $j \succeq^0 i$, $w \in \Sigma^*$ a word, and γ a successful run of $\mathcal{A}$ on w. If $\mathsf{num}_\gamma(i, a) > n$ then there exists a successful run γ' on w of $\mathcal{A}$ such that $\sigma(\gamma') < \sigma(\gamma)$.

Proof. We may assume $i < j$. The proof is analogous when $i > j$.

[6] Lemma 2.3.6 does not imply $\mathsf{loops}(t) \supseteq \mathsf{loops}(p)$ which is the only "nonreversible" difference between Lemma 2.3.4 and Lemma 2.3.6. However, it is trivial to prove it in our case, since the word ω_p is read in state r, which using Proposition 2.2.12 implies $\mathsf{loops}(r) \supseteq \mathsf{loops}(p)$ and then using $\mathsf{loops}(t) \supseteq \mathsf{loops}(r)$ by Lemma 2.3.6 we have $\mathsf{loops}(t) \supseteq \mathsf{loops}(p)$.

Let $w = v_1 v_2 v_3 v_4$ with $v_1, \ldots, v_4 \in \Sigma^*$, where v_1 is the shortest prefix of the word w such that $\mathsf{dest}_\gamma(v_1) = i$, $v_1 v_2$ is the shortest prefix of w such that $\#_a(v_2) = j - i$, and $v_1 v_2 v_3$ is the longest prefix of w such that $\mathsf{dest}_\gamma(v_1 v_2 v_3) = j$. As $\mathsf{num}_\gamma(i, a) > n$, we have $\mathsf{dest}_\gamma(v_1 v_2) = i$, which implies that $i \xrightarrow{v_2} i$. Let $\gamma_{v_1}, \gamma_{v_2}, \gamma_{v_3}, \gamma_{v_4}$ be the subruns of γ on words v_1, v_2, v_3, v_4, respectively, which together form the run γ.

We construct a run γ'_{v_2} on the word v_2 from the state i to j which uses a sequel transition whenever possible. Since $\#_a(v_2) = j - i$, the run ought to end in j. For every state $k \in [\![i, j]\!]$, we have $\mathsf{loops}(k) \supseteq \mathsf{loops}(i)$ by $j \succeq i$. Moreover, since $i \xrightarrow{v_2} i$, we have $v_2 \in (\mathsf{loops}(i) \cup \{a\})^*$, which means that a run γ'_{v_2} exists since it is possible to read any letter of v_2 in any state between i and j (except the letters a read by γ'_{v_2} on sequel transitions). Since γ'_{v_2} does not use any a-loop transitions, all sequel transitions have a weight of zero (by the normal form), and all loops on a letter different than a have zero weight, we have $\sigma(\gamma'_{v_2}) = 0$.

From existence of run γ_{v_3}, we have $i \xrightarrow{v_3} j$ and therefore $v_3 \in (\{a\} \cup \mathsf{loops}(i) \cup \mathsf{loops}(i+1) \cup \ldots \cup \mathsf{loops}(j))^* \overset{(1)}{\subseteq} (\mathsf{loops}(j) \cup \{a\})^*$. Here, (1) holds since $j \succeq i$. Therefore, $j \xrightarrow{v_3} j$ and we can construct a run γ'_{v_3} from j to j on v_3. By $j \succeq^0 i$ and other assumptions, we have that $\sigma(j, c, j)$ is not positive for any $c \in \Sigma$, which implies $\sigma(\gamma'_{v_3}) = 0$.

$$\begin{array}{l} \gamma \to \\ w = \ldots a \overbrace{\underbrace{\Gamma^*}_{i} a \underbrace{\Gamma^*}_{i+1} a \underbrace{\Gamma^* \ldots \Gamma^*}_{\ldots} a \underbrace{\Gamma^*}_{j-1} a \, \Gamma^* \ldots \Gamma^*}^{i} \underbrace{a \overbrace{\Gamma^* \ldots \Gamma^*}^{i+1} a \overbrace{\Gamma^* \ldots \Gamma^*}^{i+2} a \overbrace{\Gamma^* \ldots \Gamma^*}^{\ldots} a \overbrace{\Gamma^* \ldots \Gamma^*}^{j-1} a \overbrace{\Gamma^* \ldots \Gamma^*}^{j}}_{j} a \ldots \\ \gamma' \to \end{array}$$

Figure 2.11: By $i, i+1, \ldots, j$ we denote the states in which the runs γ or γ' are located after reading a particular part of w. We put $\Gamma = \Sigma - \{a\}$.

Now, we construct a run γ' by joining the run γ_{v_1} from 1 to i on v_1, the run γ'_{v_2} from i to j on v_2, the run γ'_{v_3} from j to j on v_3, and the run γ_{v_4} from j to n. The relationship between γ and γ' is illustrated in Figure 2.11. Then,

$$\begin{aligned} \sigma(\gamma') &= \sigma(\gamma_{v_1}) + \sigma(\gamma'_{v_2}) + \sigma(\gamma'_{v_3}) + \sigma(\gamma_{v_4}) \\ &\overset{(1)}{=} \sigma(\gamma_{v_1}) + \sigma(\gamma_{v_4}) \\ &\overset{(2)}{<} \sigma(\gamma_{v_1}) + \sigma(\gamma_{v_2}) + \sigma(\gamma_{v_3}) + \sigma(\gamma_{v_4}) \\ &= \sigma(\gamma). \end{aligned}$$

Here, (1) holds because $\sigma(\gamma'_{v_2}) = \sigma(\gamma'_{v_3}) = 0$ as we proved above and (2) holds because $\mathsf{num}_\gamma(i, a) > n$ and $\sigma(i, a, i) > 0$. $\square$

Corollary 2.3.13. Let Σ be an alphabet, $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over $\mathbb{N}_{min}$ and Σ which has positive loops only on the letter a and for every state $p \in [\![1, n]\!]$ with a positive a-loop, there exists a state $q \in [\![1, n]\!]$ with an a-loop such that $q \succeq^0 p$. Then, for every word $w \in \Sigma^*$ and every minimal run γ on w of $\mathcal{A}$,

$$\sigma(\gamma) \leq n^2 D$$

where D is the maximum of all loop transition weights.

Proof. By contradiction. Let w be a word and γ a minimal run such that $\sigma(\gamma) > n^2 D$. Since $\mathcal{A}$ is in a normal form as was assumed earlier (Remark 2.1.5) and $\mathcal{A}$ has a positive weight only on the letter a, we may conclude that there exists a state p whose a-loop was used more than n times in the run γ. By Lemma 2.3.12, we can construct a run γ' on w such that $\sigma(\gamma') < \sigma(\gamma)$, which is a contradiction with the assumption that γ is minimal. □

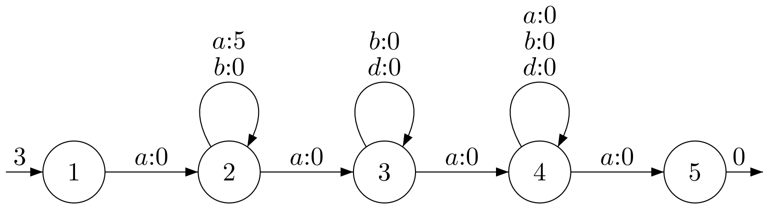

Figure 2.12: Branchless automaton $\mathcal{A}$ over $\mathbb{N}_{min}$ and $\{a, b, c, d\}$, where state 2 is superseded by state 4.

Example 2.3.14. In automaton $\mathcal{A}$ from Figure 2.12, the only state with a positive a-loop is state 2. But state 2 is superseded by state 4 which means that $\sigma(\gamma) < n^2 D = 5^3$ for every minimal successful run γ of $\mathcal{A}$ by Corollary 2.3.13. We demonstrate the construction shown in Lemma 2.3.12 on the word $w = a(ab)^{100}abdaabdaa$ and a run

$$\gamma = (1, a, 2, (ab)^{100}, 2, a, 3, bd, 3, a, 4, abda, 4, a, 5)$$

on w.[7] We see that $\sigma(\gamma) = 500$. Now, we construct a run

$$\gamma' = (1, a, 2, a, 3, b, 3, a, 4, b(ab)^{98}abdaabda, 4, a, 5)$$

on w for which $\sigma(\gamma') = 0$.

Theorem 2.3.15. Deciding the upper boundedness problem for homogeneous branchless automata over $\mathbb{N}_{min}$ belongs to $\mathbf{P}$.

Proof. Let $\mathcal{A} = (Q, \sigma, \iota, \tau)$ be a homogeneous branchless automaton in the normal form over $\mathbb{N}_{min}$ and a given alphabet Σ. First, we check whether there is a positive loop on some letter other than a. If so, then the series realised by the automaton $\mathcal{A}$ is not upper bounded by Corollary 2.3.2. Otherwise, we continue and check for every state p with a

[7]We use shorter notation of runs in which we group multiple letters read in the same state.

positive a-loop, whether there exists a state q with an a-loop such that $q \succeq^0 p$. This can be done in polynomial time. If there exists a state with a positive a-loop which is not superseded with a zero loop by any other state, the series realised by the automaton is not upper bounded by Lemma 2.3.10. Otherwise, $||\mathcal{A}||$ is bounded by $n^2D + \iota(1)$, where D is the maximum of all loop transition weights because for every word $w \in \Sigma^*$ and a minimal run γ on w, $\sigma(\gamma) \leq n^2D$, as was shown in the Corollary 2.3.13. □

Theorem 2.3.16. Deciding the absolute boundedness problem for homogeneous branchless automata over $\mathbb{N}_{min}$ and $\mathbb{Z}_{min}$ belongs to **P**.

Proof. Deciding absolute boundedness over $\mathbb{N}_{min}$ is equivalent to deciding upper boundedness over $\mathbb{N}_{min}$, since a series realised by an automaton is bounded from below by zero.

Deciding absolute boundedness over $\mathbb{Z}_{min}$ is reducible to deciding upper boundedness over $\mathbb{N}_{min}$. If the automaton has a negative loop, the series realised by it is not bounded from below. Otherwise, the series is bounded from below by the sum of the sequel transitions weights, the initial weight and the terminal weight. Then, to decide absolute boundedness, it is sufficient to decide whether the given automaton is upper bounded. Negative sequel transitions or negative initial or terminal weights do not affect the upper boundedness of the automaton. □

2.4 The Universality Problem

In this section, we show that deciding a slightly modified version of the universality problem for homogeneous branchless automata over $\mathbb{N}_{min}$ belongs to **co-NP**.

Remark 2.4.1. The universality problem is normally defined as the problem of deciding whether weights of all words over a given alphabet have a weight less than a given threshold. Such a variant of the universality problem can be solved trivially for branchless automata over $\mathbb{N}_{min}$. It is sufficient to check whether a given automaton has more than one state, in which case, we can easily construct a word, which does not belong to the *support* of the series realised by a given automaton. If the automaton has only one state, an instance of the universality problem is affirmative iff the state has loops over all letters with a weight of zero. Therefore, in all following theorems, we consider the universality problem only over the *support* of a series realised by an automaton.

Remark 2.4.2. For general weighted automata, and even for acyclic automata with loops defined in the next chapter, the modified version of the universality problem is reducible to the standard universality problem and vice-versa. Below, we give an outline of the reductions between the problems.

Let $\mathcal{A}$ be a given weighted automaton over $\mathbb{N}_{min}$ and a given alphabet Σ.

Tge instance of the modified universality problem with a threshold $T \in \mathbb{N}$ and the automaton $\mathcal{A}$ can be reduced to the instance of the standard universality problem with the threshold T and an automaton $\mathcal{B}$, which we construct as follows. The automaton $\mathcal{B}$ is constructed as a disjoint union of the automaton $\mathcal{A}$ and an automaton which realises the complement of $\mathcal{A}$ with all weights set to zero.

The instance of the standard universality problem with a threshold $T \in \mathbb{N}$ and the automaton $\mathcal{A}$ can be reduced to the instance of the modified universality problem with the threshold T and the automaton $\mathcal{A}$, to which we add a state with the initial weight of T, the terminal weight of zero, and loop transitions over all letters from Σ with the weight of zero.

Now, we prove multiple lemmas that enable us to restrict the language of words we need to consider when deciding the universality problem.

Lemma 2.4.3. Let Σ be an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over $\mathbb{N}_{min}$ and Σ which has positive loops only on the letter a. Let $T \in \mathbb{N}$. Then, there exists a word $w \in \Sigma^*$ such that $(||\mathcal{A}||, w) = T$ iff there exists a word $v \in \Omega_{\mathcal{A}}$ such that $(||\mathcal{A}||, v) = T$.

Proof. Proving $\Leftarrow$ is trivial since $\Omega_{\mathcal{A}} \subseteq \Sigma^*$ and so we need to prove only $\Rightarrow$. For every $T \in \mathbb{N}$, let $w \in \Sigma^*$ be a word such that $(||\mathcal{A}||, w) = T$ and let γ_w be any of the minimal runs of $\mathcal{A}$ on the word w. We construct a word $v \in \Omega_{\mathcal{A}}$ and a successful run γ_v on v such that $\sigma(\gamma_w) = \sigma(\gamma_v)$ and we prove that γ_v is minimal on v.

Construction of v and γ_v. We decompose the word w as follows:

$$w = w_0 w_{1,1} w_{1,2} \dots w_{1,m_1} w_{2,1} \dots w_{2,m_2} \dots w_{n,1} \dots w_{n,m_n},$$

where $m_1 = \mathsf{num}_{\gamma_w}(1, a)$, $m_i = \mathsf{num}_{\gamma_w}(i, a) + 1$ for all $i \in [\![2, n]\!]$, $w_0 \in (\Sigma - \{a\})^*$ and $w_{i,j} \in a(\Sigma - \{a\})^*$ for all $i \in [\![1, n]\!]$, and every $j \in [\![1, m_i]\!]$.

In other words, $w_{i,1} w_{i,2} \dots w_{i,m_i}$ is exactly the word read in state i in γ_w (except first letter of $w_{i,1}$ which is read on a transition between state $i-1$ and i and with an exception for the state 1 which reads the word $w_0 w_{1,1} \dots w_{1,m_1}$), where each part $w_{i,j}$ contains exactly one letter a which is at the beginning of the word. The decomposition of w is unambiguous.

The word v is constructed such that

$$v = \omega_1^{num_{\gamma_w}(1,a)} \omega_2^{num_{\gamma_w}(2,a)+1} \dots \omega_n^{num_{\gamma_w}(n,a)+1}.$$

The relationship between w and v is shown in Figure 2.13.

$$
\begin{aligned}
w = {} & \overbrace{\Gamma^*}^{w_0}\ \overbrace{a\Gamma^*}^{w_{1,1}}\ \overbrace{a\Gamma^*}^{w_{1,2}}\ \overbrace{a\Gamma^*\ldots\Gamma^*}^{\ldots}\ \overbrace{a\Gamma^*}^{w_{1,m_1}}\ \overbrace{a\Gamma^*}^{w_{2,1}}\ \overbrace{a\Gamma^*\ldots\Gamma^*}^{\ldots}\ \overbrace{a\Gamma^*}^{w_{2,m_2}}\ \overbrace{a\Gamma^*}^{w_{3,1}}\ \overbrace{\ldots\Gamma^*}^{\ldots}\ \overbrace{a\Gamma^*}^{w_{n,1}}\ \overbrace{a\Gamma^*\ldots\Gamma^*}^{\ldots}\ \overbrace{a\Gamma^*}^{w_{n,m_n}} \\
& \downarrow \\
v = {} & \underbrace{a\Gamma^*}_{\omega_1}\ \underbrace{a\Gamma^*}_{\omega_1}\ \underbrace{a\Gamma^*\ldots\Gamma^*}_{\ldots}\ \underbrace{a\Gamma^*}_{\omega_1}\ \underbrace{a\Gamma^*}_{\omega_2}\ \underbrace{a\Gamma^*\ldots\Gamma^*}_{\ldots}\ \underbrace{a\Gamma^*}_{\omega_2}\ \underbrace{a\Gamma^*}_{\omega_3}\ \underbrace{\ldots\Gamma^*}_{\ldots}\ \underbrace{a\Gamma^*}_{\omega_n}\ \underbrace{a\Gamma^*\ldots\Gamma^*}_{\ldots}\ \underbrace{a\Gamma^*}_{\omega_n}
\end{aligned}
$$

Figure 2.13: The relationship between w and v where $\Gamma = \Sigma - \{a\}$.

We construct a successful run γ_v on v such that $\mathsf{num}_{\gamma_v}(i,a) = \mathsf{num}_{\gamma_w}(i,a)$ for every $i \in [\![1,n]\!]$. The run exists, since $(i-1) \xrightarrow{\omega_i} i$ holds for every $i \in [\![2,n]\!]$ and $i \xrightarrow{\omega_i} i$ holds for every $i \in [\![1,n]\!]$ such that $\mathsf{num}_{\gamma_w}(i,a) \geq 1$. By the Corollary 2.2.6 we have

$$\sigma(\gamma_w) = \sigma(\gamma_v). \tag{2.3}$$

The proof that γ_v is minimal on v. Now, we prove the minimality of γ by showing that for every successful run γ'_v on v, we can construct a successful run γ'_w on w such that $\sigma(\gamma'_w) = \sigma(\gamma'_v)$. Then, if γ_v is not a minimal run on v, γ_w is also not a minimal run on w by (2.3) which contradicts the assumption that γ_w is a minimal run on w.

For every $i \in [\![1,n]\!]$ and every $j \in [\![1,m_i]\!]$, $\mathsf{alph}(w_{i,j}) \subseteq \mathsf{loops}(i) \cup \{a\}$, because otherwise, it would not be possible to read $w_{i,j}$ in the state i in the run γ_w. Also, $\mathsf{alph}(\omega_i) = \mathsf{loops}(i) \cup \{a\}$ which means that for every state $p \in [\![1,n]\!]$, $\mathsf{loops}(p) \supseteq \mathsf{alph}(w_{i,j})$ whenever $\mathsf{loops}(p) \supseteq \mathsf{alph}(\omega_i)$. Thus, we have that for every $p, i \in [\![1,n]\!]$

$$
\begin{gathered}
\text{if } p \xrightarrow{\omega_i} p \text{ then } p \xrightarrow{w_{i,j}} p \text{ for every } j \in [\![1,m_i]\!], \\
\text{if } p \xrightarrow{\omega_i} p+1 \text{ then } p \xrightarrow{w_{i,j}} p+1 \text{ for every } j \in [\![1,m_i]\!]
\end{gathered}
\tag{2.4}
$$

(the second implication holds only for $p \in [\![1,n-1]\!]$).

Let γ'_v be a successful run on v. We want to prove that there exists a run γ'_w on w such that

$$\mathsf{num}_{\gamma'_w}(i,a) = \mathsf{num}_{\gamma'_v}(i,a) \text{ for every } i \in [\![1,n]\!] \tag{2.5}$$

Firstly, $1 \xrightarrow{w_0} 1$ follows obviously from the existence of the run γ_w.

For every $i \in [\![1,n]\!]$, let $p, q \in [\![1,n]\!]$ be states such that γ'_v reads the subword $\omega_i^{\mathsf{num}_{\gamma_w}(i,a)+1}$ of v from p to q.[8] (In the case of $i = 1$, we use the subword $\omega_1^{\mathsf{num}_{\gamma_w}(1,a)}$). Then in order for (2.5) to hold, γ'_w needs to read $w_{i,1} \ldots w_{i,m_i}$ from p to q and use the sequel and loop transitions in the same way as γ'_v does. For every state $r \in [\![p,q]\!]$, if γ'_v reads $r \xrightarrow{\omega_i} r+1$, then by (2.4) we have $r \xrightarrow{w_{i,j}} r+1$ for every $j \in [\![1,m_i]\!]$. Similarly

[8] Here, the subword $\omega_i^{\mathsf{num}_{\gamma_w}(i,a)+1}$ is not arbitrary, but the one, which follows after the word $\omega_1^{num_{\gamma_w}(1,a)} \ldots \omega_{i-1}^{num_{\gamma_w}(i-1,a)+1}$.

if γ'_v reads $r \xrightarrow{\omega_i} r$, then $r \xrightarrow{w_{i,j}} r$ holds for every $j \in [\![1, m_i]\!]$. Therefore it is possible to read $w_{i,1} \dots w_{i,m_i}$ from p to q while using the sequel and loop transitions in the same way as γ'_v does, which concludes that the run γ'_w exists. By Corollary 2.2.6 we have $\sigma(\gamma'_v) = \sigma(\gamma'_w)$ and hence our proof is complete. $\square$

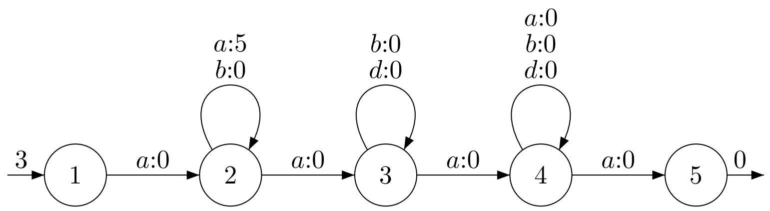

Figure 2.14: Branchless automaton $\mathcal{A}$ over $\mathbb{N}_{min}$ and $\{a, b, c, d\}$, where state 2 is superseded by state 4.

Example 2.4.4. Let $\mathcal{A}$ be the automaton of Figure 2.14, $w = abbaadbadda$ a word and

$$\gamma_w = (1, a, 2, bb, 2, a, 3, a, 4, dbadd, 4, a, 5)$$

a minimal run on w. Based on Lemma 2.4.3 we decompose w as follows:

$$m_1 = \mathsf{num}_{\gamma_w}(1, a) = 0, \quad m_2 = \mathsf{num}_{\gamma_w}(2, a) + 1 = 1, \quad m_3 = \mathsf{num}_{\gamma_w}(3, a) + 1 = 1,$$
$$m_4 = \mathsf{num}_{\gamma_w}(4, a) + 1 = 2, \quad m_5 = \mathsf{num}_{\gamma_w}(5, a) + 1 = 1,$$
$$w_0 = \varepsilon, w_{2,1} = abb, w_{3,1} = a, w_{4,1} = adb, w_{4,2} = add, w_{5,1} = a.$$

We construct a word $v = \omega_1^0 \omega_2^1 \omega_3^1 \omega_4^2 \omega_5^1 = ab\ abd\ abd\ abd\ a$.

We might easily see that for every run γ'_v on v we can construct a run γ'_w on w such that $\mathsf{num}_{\gamma'_w}(i, a) = \mathsf{num}_{\gamma'_v}(i, a)$ for every $i \in [\![1, 5]\!]$, and therefore $\sigma(\gamma'_w) = \sigma(\gamma'_v)$. For example, if the word $\omega_2 = ab$ is read in state 4 by γ'_v, then also $w_{2,1} = abb$ may be read in state 2 by γ'_w. If $\omega_4 = abd$ is read in state 3 by γ'_v, then also $w_{4,1} = adb$ and $w_{4,2} = add$ may be read in state 3 by γ'_w.

The minimal run γ_v on v is constructed using a minimal run γ_w on w such that $\mathsf{num}_{\gamma_w}(i, a) = \mathsf{num}_{\gamma_v}(i, a)$ for every $i \in [\![1, 5]\!]$:

$$\gamma_v = (1, a, 2, b, 2, a, 3, bd, 3, a, 4, bdabd, 4, a, 5).$$

Now we are about to prove a lemma which tells us that every minimal run on a word $u(aw)^k$ from some state p to some state q, with $a \notin \mathsf{alph}(w)$, uses only a-loops with a minimum weight of all a-loop weights between p and q.

Lemma 2.4.5. Let Σ be an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ a homogeneous branchless automaton over $\mathbb{N}_{min}$ and Σ which has positive loops only on the letter a. Let also

$u, w \in (\Sigma - \{a\})^*$, $k \in \mathbb{N}$ be such that $k \geq n$ and γ be a minimal run on $u(aw)^k$ from some state $p \in [\![1, n]\!]$ to some state $q \in [\![p, n]\!]$. Let $C \in \mathbb{N}$ be the smallest number such that there exists a state $r \in [\![p, q]\!]$ for which $\sigma(r, a, r) = C$. Then $\sigma(\gamma) = (k - q + p)C$, $\mathsf{loops}(p) \supseteq \mathsf{alph}(u)$ and for every state $r \in [\![p + 1, q]\!]$, $\mathsf{loops}(r) \supseteq \mathsf{alph}(w)$ holds.

Proof. Since $p \overset{u(aw)^k}{\to} q$, the word w must be read on every state r between p and q (possibly except the state p where u is read) and thus we can prove, similarly as in Proposition 2.2.20, that $\mathsf{loops}(r) \supseteq \mathsf{alph}(w)$ for every $r \in [\![p + 1, q]\!]$ and $\mathsf{loops}(p) \supseteq \mathsf{alph}(u)$.

Since $u(aw)^k$ contains $\#_a(u(aw)^k) = k$ times the letter a and the run γ uses $q - p$ sequel transitions, therefore γ uses $k - q + p$ times some a-loop. Since $k \geq n$ and $q - p < n$, we have that $k - q + p > 0$, which proves the existence of a state $r \in [\![p, q]\!]$ such that $\sigma(q, a, q) = C$. Also, since the automaton is in the normal form and every loop on a letter different from a has zero weight, the weight of the run γ is a sum of a-loop weights similarly as in Corollary 2.2.6. Therefore

$$\sigma(\gamma') \geq (k - q + p)C \tag{2.6}$$

hold for every run γ' on $u(aw)^k$ from p to q, Let r be some state with $\sigma(r, a, r) = C$. Then we can construct a run γ'' from p to q on $u(aw)^k$ first reading $u(aw)^{r-p}$ to reach the state r, then reading $(aw)^{k-q+p}$ on r and finally reading $(aw)^{q-r}$ to reach the state q.

By (2.6) we see that γ'' is a minimal run on $u(aw)^k$ from p to q of $\mathcal{A}$ and therefore $\sigma(\gamma) = \sigma(\gamma'') = (k - q + p)C$. □

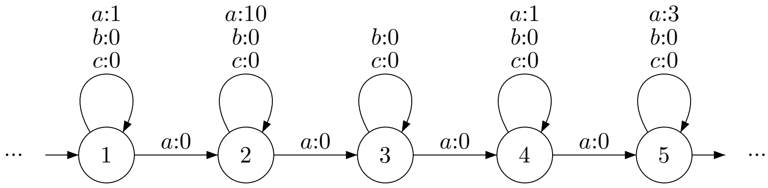

Figure 2.15: Branchless automaton $\mathcal{B}$ over $\mathbb{N}_{min}$ and $\{a, b, c\}$.

Example 2.4.6. Let $\mathcal{B}$ be the automaton from Figure 2.15, $u = bc$, $w = cbc$ and we want to find out, what is the weight of a minimal run from state 1 to state 5 on $u(aw)^7$. We easily see that $C = 1$ is the minimum of a-loop weights between states 1 and 5. Then, the minimal run γ on $u(aw)^7$ from 1 to 5 must use exactly 3 times some a-loop with a weight of 1. Therefore the only possible minimal runs are

$$\gamma_1 = (1, bcacbcacbcacbc, 1, a, 2, cbc, 2, a, 3, cbc, 3, a, 4, cbc, 4, a, 5, cbc, 5),$$

$$\gamma_2 = (1, bcacbcacbc, 1, a, 2, cbc, 2, a, 3, cbc, 3, a, 4, cbcacbc, 4, a, 5, cbc, 5),$$

$$\gamma_3 = (1, bcacbc, 1, a, 2, cbc, 2, a, 3, cbc, 3, a, 4, cbcacbcacbc, 4, a, 5, cbc, 5),$$

$$\gamma_4 = (1, bc, 1, a, 2, cbc, 2, a, 3, cbc, 3, a, 4, cbcacbcacbcacbc, 4, a, 5, cbc, 5),$$

and for each one $\sigma(\gamma_1) = \sigma(\gamma_2) = \sigma(\gamma_3) = \sigma(\gamma_4) = 3$.

Lemma 2.4.7. Let Σ be an alphabet and $\mathcal{A} = (n, \sigma, \iota, \tau)$ an upper bounded homogeneous branchless automaton over $\mathbb{N}_{min}$ and Σ. For every word $w \in \Omega_{\mathcal{A}}$ there exists a word $v = \omega_1^{l_1} \dots \omega_n^{l_n}$ with $l_1, \dots, l_n \in [\![1, n^2 + n]\!]$ such that $(||\mathcal{A}||, w) = (||\mathcal{A}||, v)$.

Proof. Let $w = \omega_1^{k_1} \dots \omega_n^{k_n}$ be a word where $k_1, \dots, k_n \in \mathbb{N}$. We construct a word v such that $v = \omega_1^{l_1} \dots \omega_n^{l_n}$ where for every $i \in [\![1, n]\!]$,

$$l_i = \begin{cases} k_i & \text{if } k_i \leq n^2 + n, \\ n^2 + n & \text{otherwise.} \end{cases}$$

Now, we prove $(||\mathcal{A}||, w) = (||\mathcal{A}||, v)$ by showing that for every minimal run γ_w on w we can find a run γ_v on v such that $\sigma(\gamma_w) = \sigma(\gamma_v)$ and vice-versa.

$\Rightarrow$ Let γ_w be a minimal run on w. For every $i \in [\![1, n]\!]$ such that $k_i > n^2 + n$, let $p_i = \mathsf{dest}_{\gamma_w}(\omega_1^{k_1} \dots \omega_{i-1}^{k_{i-1}} a)$, $q_i = \mathsf{dest}_{\gamma_w}(\omega_1^{k_1} \dots \omega_i^{k_i})$ and $u_i \in (\Sigma - \{a\})^*$ such that $\omega_i = a u_i$. In other words, γ_w reads the word $u_i \omega_i^{k_i}$ from state p_i to state q_i.

Since γ_w is a minimal run, then also a subrun $\gamma_{w,i}$ of run γ_w from p_i to q_i on $u_i \omega_i^{k_i}$ is minimal by Proposition 2.2.16. By Lemma 2.4.5, the weight $\sigma(\gamma_{w,i})$ depends on the smallest weight of all a-loops between p_i and q_i. Let $r_i \in [\![p_i, q_i]\!]$ be any of the states with the smallest weight of an a–loop between p_i and q_i. If $\sigma(r_i, a, r_i) > 0$, then for every state $t \in [\![p_i, q_i]\!]$, $\sigma(t, a, t) \geq \sigma(r_i, a, r_i) > 0$ holds which means that $\gamma_{w,i}$ reads letters a only in sequel transitions or in states with positive a-loops. But since $\#_a(u_i(aw_i)^{k_i}) = k_i > n^2 + n$ and the number of sequel transitions is less than n, the minimal run $\gamma_{w,i}$ would have to use a-loops more than n^2 times and therefore use some single a-loop more than n times (since the number of states is n) which is a contradiction with the upper boundedness as it was shown in Lemma 2.3.12. Therefore $\sigma(r_i, a, r_i) = 0$, from which we have $\sigma(\gamma_{w,i}) = 0$.

Now, we construct the run γ_v. For every $i \in [\![1, n]\!]$ if $k_i \leq n^2 + n$, then $k_i = l_i$ and γ_v reads the word $\omega_i^{l_i}$ in the same way as γ_w reads $\omega_i^{k_i}$. If $k_i > n^2 + n$, γ_v reads $\omega_i^{l_i} = \omega_i^{n^2+n}$ such that

$$p_i \xrightarrow{\omega_i^{r_i - p_i}} r_i \xrightarrow{\omega_i^{n^2+n-(q_i-p_i)}} r_i \xrightarrow{\omega_i^{q_i - r_i}} q_i.$$

Such a run is possible because by Lemma 2.4.5 we have $\mathsf{loops}(s) \supseteq \mathsf{alph}(\omega_i) - \{a\} = \mathsf{loops}(i)$ for every state $s \in [\![p_i + 1, q_i]\!]$.

Now, we show that $\sigma(\gamma_v) = \sigma(\gamma_w)$, which holds because for every $i \in [\![1, n]\!]$, if $k_i > n^2 + n$, then the weight of the word $\omega_i^{k_i}$ in γ_w is zero as was proved above (we proved that $u_i(aw_i)^{k_i}$ has zero weight in γ_w and u_i does not add to weight since it does not include the letter a) and the same holds for word $\omega_i^{l_i}$ on the run γ_v, because it uses states with zero weight a-loops. If $k_i \leq n^2+n$, the weight of the word $\omega_i^{k_i}$ for γ_w is the same as the word $\omega_i^{l_i}$ in γ_v, because the runs are identical in these parts.

$\Leftarrow$ From a successful run γ_v, we construct the run γ_w as follows. For every $i \in [\![1,n]\!]$, if $k_i > n^2+n$, then there must exist a state r_i with the same properties as shown in the previous part. In the run γ_w, the word $\omega_i^{k_i}$ is read such that

$$p_i \xrightarrow{\omega_i^{r_i-p_i}} r_i \xrightarrow{\omega_i^{k_i-(q_i-p_i)}} r_i \xrightarrow{\omega_i^{q_i-r_i}} q_i.$$

If $k_i \leq n^2+n$, then $\omega_i^{l_i} = \omega_i^{k_i}$ and the runs γ_w and γ_v on these words are identical. Similarly, to the above, we can prove that $\sigma(\gamma_v) = \sigma(\gamma_w)$. □

Theorem 2.4.8. The universality problem[9] for homogeneous branchless automata over $\mathbb{N}_{min}$ is decidable in **co-NP** time.

Proof. Let $k \in \mathbb{N}$ be a threshold of the universality problem. First, we check upper boundedness which is done in polynomial time as was proved in Theorem 2.3.16. We know that if there exists a word $w \in \Sigma^*$, such that $(||\mathcal{A}||, w) > k$, then by Lemma 2.4.3, there exists a word $v \in \Omega_{\mathcal{A}}$ such that $(||\mathcal{A}||, v) = (||\mathcal{A}||, w)$ and by Lemma 2.4.7, there exists a word $u = \omega_1^{l_1} \dots \omega_n^{l_n}$ with $l_1, \dots, l_n \leq n^2+n$ such that $(||\mathcal{A}||, u) = (||\mathcal{A}||, v)$. Therefore, to decide the complement of universality, it is sufficient to check whether there exists a word $u = \omega_1^{l_1} \dots \omega_n^{l_n}$ with $l_1, \dots, l_n \leq n^2+n$ such that $(||\mathcal{A}||, u) > k$, which is possible in polynomial time using a nondeterministic Turing machine. Therefore the problem belongs to **co-NP**. □

Theorem 2.4.9. The $\forall$-exact problem for homogeneous branchless automata over $\mathbb{N}_{min}$ and $\mathbb{Z}_{min}$ is decidable in **co-NP** time.

Proof. Deciding $\forall$-exact over $\mathbb{N}_{min}$ is reducible to the universality problem (the statement (ii) of Proposition 1.5.1).

For automata over $\mathbb{Z}_{min}$, the answer to the instance of the problem is negative, if a given automaton has a negative loop. Otherwise, we can obtain a new automaton with nonnegative weights over $\mathbb{N}_{min}$ for which the instance of the $\forall$-exact problem is affirmative iff it is affirmative for the given automaton [2, p. 16]. □

[9]The modified version over the *support* of the realised series.

2.5 The $\exists$-exact Problem

The only decision problem considered in [2] which (provided $\mathbf{P} \neq \mathbf{NP}$) does not belong to $\mathbf{P}$ in the deterministic case is the $\exists$-exact problem. In [2], the authors proved the $\mathbf{NP}$-hardness of this problem using a reduction of the subset-sum problem, known to be $\mathbf{NP}$-hard [8], to the $\exists$-exact problem for an acyclic automaton with loops. Here, we are about to strengthen the reduction by using an automaton with only one state, which proves the $\mathbf{NP}$-hardness of the $\exists$-exact problem even for branchless and homogeneous branchless automata. The reduction is inspired by a reduction of the subset-sum problem to the unbounded knapsack problem demonstrated by K. A. Hansen [9].

Theorem 2.5.1. The $\exists$-exact problem for weighted automata with one state over the tropical semiring $\mathbb{N}_{min}$ is $\mathbf{NP}$-hard.

Proof. Our proof uses a reduction from the subset-sum problem, which is known to be $\mathbf{NP}$-hard [8]. The subset-sum problem asks for a given $a_1, \ldots, a_m \in \mathbb{N}$ with $m \in \mathbb{N}$ and a desired sum $A \in \mathbb{N}$, whether there exist $x_1, \ldots, x_m \in \{0, 1\}$ such that $\sum_{i=1}^{m} x_i a_i = A$. We put $d = 1 + max\{a_1, \ldots, a_m\}$. We may assume $A < md$, since otherwise, $A > \sum_{i=1}^{m} a_i$ which would make the answer to the instance of the subset-sum problem negative.

We reduce this problem to the $\exists$-exact problem for a weighted automaton with one state $\mathcal{A} = (1, \sigma, \iota, \tau)$ over $\mathbb{N}_{min}$ and Σ where

$$\begin{aligned}
\Sigma &= \{c_1^0, c_1^1, c_2^0, c_2^1, , \ldots, c_m^0, c_m^1\}, \\
\iota(1) &= \tau(1) = 0, \\
\sigma(p, c_i^0, p) &= (2^{m+1} + 2^i)md \text{ for every } i \in [\![1, m]\!] \\
\sigma(p, c_i^1, p) &= (2^{m+1} + 2^i)md + a_i \text{ for every } i \in [\![1, m]\!].
\end{aligned}$$

The intuitive meaning of the alphabet Σ is that letter c_i^1 represents $x_i = 1$ and letter c_i^0 represents $x_i = 0$.

Now, consider an instance of the $\exists$-exact problem, in which we ask whether there exists a word $w \in \Sigma^*$ such that $(||\mathcal{A}||, w) = C$, where

$$C = (m2^{m+1} + 2^m + \ldots + 2^1)md + A.$$

First, we want to show that for a word $w \in \Sigma^*$, $(||\mathcal{A}||, w) = C$ holds iff the following conditions are met:

(i) w contains exactly one letter from $\{c_i^0, c_i^1\}$ for each $i \in [\![1, m]\!]$,

(ii) $\sum_{i=1}^{m} \#_{c_i^1}(w) a_i = A$.

- $\Rightarrow$ First, we prove that $|w| \leq m$. If $|w| > m$, then $(||\mathcal{A}||, w) \geq (m+1)(2^{m+1}md + 2md) = 2^{m+1}m^2d + 2^{m+1}md + 2m^2d + 2md > 2^{m+1}m^2d + (2^m + \ldots + 2^1)md + 2md > (2^{m+1}m + 2^m + \ldots + 2^1)md + A = C$ which is a contradiction with $(||\mathcal{A}||, w) = C$.

  Let $w = c_{k_1}^{l_1} c_{k_2}^{l_2} \ldots c_{k_j}^{l_j}$ with $j \in [\![1, m]\!]$, $k_1, \ldots, k_j \in [\![1, m]\!]$ and $l_1, \ldots, l_j \in \{0, 1\}$. Then

$$(||\mathcal{A}||, w) = \sum_{i=1}^{j}((2^{m+1} + 2^{k_i})md + l_i a_{k_i}) = \sum_{i=1}^{j}(2^{m+1} + 2^{k_i})md + \sum_{i=1}^{j} l_i a_{k_i} = C. \tag{2.7}$$

  Here, $\sum_{i=1}^{j} l_i a_{k_i} < md$ since $j \leq m$ and $a_{k_i} \leq d$ for every $i \in [\![1, j]\!]$. Also, $A < md$ by the assumptions. Hence, if we take (2.7) modulo md we get

$$\sum_{i=1}^{j} l_i a_{k_i} = A, \tag{2.8}$$

  which necessarily implies

$$\begin{gathered}\sum_{i=1}^{j}(2^{m+1} + 2^{k_i})md = (m2^{m+1} + 2^m + \ldots + 2^1)md \\ \sum_{i=1}^{j}(2^{m+1} + 2^{k_i}) = (m2^{m+1} + 2^m + \ldots + 2^1).\end{gathered} \tag{2.9}$$

  By $j \leq m$ and from the binary representation of $(m2^{m+1} + 2^m + \ldots + 2^1)$ we clearly see, that (2.9) holds iff $\{k_1, \ldots, k_j\} = \{1, \ldots, m\}$ which proves (i). But then, (2.8) implies (ii).

- $\Leftarrow$ Let $w = c_{k_1}^{l_1} c_{k_2}^{l_2} \ldots c_{k_j}^{l_j}$ with $j \in [\![1, n]\!]$, $k_1, \ldots, k_j \in [\![1, m]\!]$ and $l_1, \ldots, l_j \in \{0, 1\}$. By (i), we have $\sum_{i=1}^{j}(2^{m+1} + 2^{k_i})md = (m2^{m+1} + 2^m + \ldots + 2^1)md$ and by (i) and (ii) we have $\sum_{i=1}^{j} l_i a_i = A$ which together implies $(||\mathcal{A}||, w) = C$.

We proved that if there exists a word $w \in \Sigma^*$ such that $(||\mathcal{A}||, w) = C$, then by (i) and (ii), there exists $l_1, ..., l_m \in \{0, 1\}$ such that $\sum_{i=1}^{m} l_i a_i = A$ which is a solution to the instance of the subset-sum problem. Also, if there exists $x_1, ..., x_m \in \{0, 1\}$ such that $\sum_{i=1}^{m} x_i a_i = A$, then we can construct a word $w = c_1^{x_1} ... c_m^{x_m}$ for which (i) and (ii) holds, which implies $(||\mathcal{A}||, w) = C$.

Therefore, we proved that the presented instance of the ∃-exact problem is affirmative iff the given instance of the subset-sum problem is affirmative and therefore the ∃-exact problem for the presented one-state weighted automata is **NP**-hard since the subset-sum problem is **NP**-complete. □

Corollary 2.5.2. The ∃-exact problem for homogeneous branchless automata and branchless automata over $\mathbb{N}_{min}$ and $\mathbb{Z}_{min}$ is **NP**-hard.

Deciding the $\exists$-exact problem for deterministic weighted automata belongs to **NP** [2], from which we conclude the following Corollary.

Corollary 2.5.3. The $\exists$-exact problem for deterministic homogeneous branchless automata, deterministic branchless automata, and deterministic acyclic automata with loops over $\mathbb{N}_{min}$ and $\mathbb{Z}_{min}$ is **NP**-complete.

Chapter 3

Acyclic Automata with Loops

In the previous chapter, we have seen that some decision problems, e.g., upper boundedness or universality, improved their time complexity for a structurally restricted class of weighted automata – namely branchless automata. Our aim now is to study another structurally restricted class of automata – acyclic automata with loops.

For this class of automata, we define a normal form called branchless normal form, and prove that series realised by an acyclic automaton with loops is equal to series realised by some automaton in the branchless normal form.

We also show that every series realised by an acyclic automaton with loops can be expressed as a finite sum of series realised by branchless automata.

Finally, we examine the hardness of selected decision problems over tropical semirings $\mathbb{N}_{min}$ and $\mathbb{Z}_{min}$ for which we prove that none of the problems which is decidable for general weighted automata becomes easier for acyclic weighted automata with loops.

Acyclic automata with loops can be equivalently described as automata which do not have cycles longer than one, or likewise, as automata where we never return to some state once we leave it. Alternative names include partially ordered automata [13], extensive automata [16], or acyclic automata [11]. In all mentioned cases, they have been studied in the unweighted context.

Definition 3.0.1. Let S be a semiring and Σ an alphabet. *An acyclic weighted automaton with loops* is a weighted automaton $\mathcal{A} = (Q, \sigma, \iota, \tau)$ over S and Σ where for every run $\gamma = (q_1, a_1, q_2, \ldots, q_{m-1}, a_{m-1}, q_m)$ of $\mathcal{A}$, where $q_1, \ldots, q_m \in Q$ and $a_1, \ldots, a_{m-1} \in \Sigma$, $q_1 = \ldots = q_m$ whenever $q_1 = q_m$.

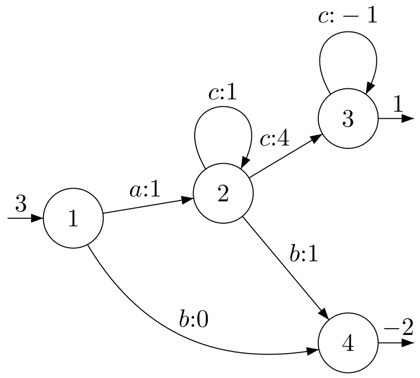

Figure 3.1: An acyclic automaton with loops $\mathcal{A}$.

3.1 The Branchless Normal Form

This section proves the normal form theorem for acyclic weighted automata with loops. The normal form "splits" all branches of a given automaton into separate branchless automata.

Definition 3.1.1. Let S be a semiring, Σ an alphabet and $\mathcal{A} = (Q, \sigma, \iota, \tau)$ an acyclic weighted automaton with loops over S and Σ. A run $\gamma = (q_1, a_1, q_2 \dots, q_{m-1}, a_{m-1}, q_m)$ of $\mathcal{A}$ with $m \in \mathbb{N}^+$, $q_1, \dots, q_m \in Q$ and $a_1, \dots, a_{m-1} \in \Sigma$ has *nonrepeating states* if $q_i \neq q_{i+1}$ holds for every $i \in [\![1, m-1]\!]$.

Proposition 3.1.2. Let S be a semiring, Σ an alphabet and $\mathcal{A} = (Q, \sigma, \iota, \tau)$ an acyclic weighted automaton with loops over S and Σ. There is only a finite number of runs of $\mathcal{A}$, which have nonrepeating states.

Proof. Let $\gamma = (q_1, a_1, q_2, \dots, q_{m-1} a_{m-1}, q_m)$ be a run of $\mathcal{A}$ with nonrepeating states where $m \in \mathbb{N}^+$, $q_1, \dots, q_m \in Q$ and $a_1, \dots, a_{m-1} \in \Sigma$. By the definition of acyclic automata with loops, if $q_i = q_j$ for some $i, j \in [\![1, m]\!]$, then $q_i = \dots = q_j$, which would be a contradiction with γ having nonrepeating states. Therefore $q_i \neq q_j$ for every $i, j \in [\![1, m]\!]$ such that $i \neq j$, which implies that $|\gamma| \leq |Q|$.

Hence we proved that $|\gamma| \leq |Q|$ for every run γ with nonrepeating states and since Σ is finite, there is only a finite number of runs with nonrepeating states. $\square$

Definition 3.1.3. Let S be a semiring and Σ an alphabet. An acyclic weighted automaton with loops $\mathcal{A} = (Q, \sigma, \iota, \tau)$ is in *branchless normal form* if there exists $B \in \mathbb{N}^+$ and branchless automata $\mathcal{A}_1, \dots, \mathcal{A}_B$ over S and Σ such that $\mathcal{A} = \mathcal{A}_1 + \dots + \mathcal{A}_B$.

Theorem 3.1.4. Let S be a semiring and Σ an alphabet. For every acyclic weighted automaton with loops $\mathcal{A} = (n, \sigma_{\mathcal{A}}, \iota_{\mathcal{A}}, \tau_{\mathcal{A}})$ over S and Σ there exists an acyclic weighted automaton with loops $\mathcal{B} = (Q_{\mathcal{B}}, \sigma_{\mathcal{B}}, \iota_{\mathcal{B}}, \tau_{\mathcal{B}})$ over S and Σ in branchless normal form such that $||\mathcal{A}|| = ||\mathcal{B}||$.

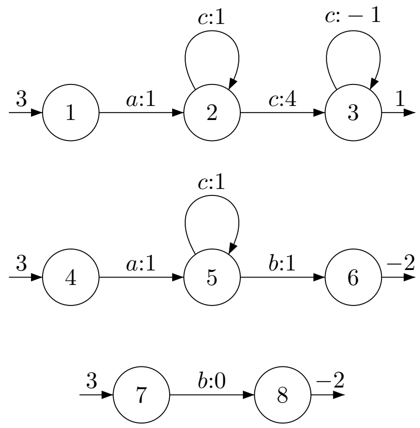

Figure 3.2: Automaton from Figure 3.1 in the branchless normal form.

Proof. First, we construct a set of branchless automata. For every successful run with nonrepeating states $\gamma = (q_1, a_1, q_2, \ldots, q_{m-1}, a_{m-1}, q_m)$ of $\mathcal{A}$ where $m \in \mathbb{N}^+$, $q_1, \ldots, q_m \in [\![1, n]\!]$, $a_1, \ldots, a_{m-1} \in \Sigma$, we construct a branchless automaton $\mathcal{A}_\gamma = (Q_\gamma, \sigma_\gamma, \iota_\gamma, \tau_\gamma)$ over S and Σ where

$$\begin{aligned} Q_\gamma &= \{p_{\gamma,q_1}, \ldots, p_{\gamma,q_m}\} \\ \sigma_\gamma(p_{\gamma,q_i}, a_i, p_{\gamma,q_{i+1}}) &= \sigma_{\mathcal{A}}(q_i, a_i, q_{i+1}) \text{ for every } i \in [\![1, m-1]\!] \\ \sigma_\gamma(p_{\gamma,q_i}, c, p_{\gamma,q_i}) &= \sigma_{\mathcal{A}}(q_i, c, q_i) \text{ for every } i \in [\![1, m]\!] \text{ and every } c \in \Sigma \\ \iota_\gamma(p_{\gamma,q_1}) &= \iota_{\mathcal{A}}(q_1) \\ \tau_\gamma(p_{\gamma,q_m}) &= \tau_{\mathcal{A}}(q_m) \end{aligned}$$

and every other value of $\sigma_\gamma, \iota_\gamma$ and τ_γ is equal to 0.[1] The automaton $\mathcal{A}_\gamma$ is obviously branchless. Then, we construct an automaton $\mathcal{B}$ over S and Σ in a branchless normal form as a disjoint union of automata $\mathcal{A}_\gamma$ over all successful run γ with nonrepeating states. The union is finite since there is only a finite number of successful runs with nonrepeating states by Proposition 3.1.2.

Now, we are about to prove the Proposition by showing a weight preserving bijection between successful runs of $\mathcal{A}$ and $\mathcal{B}$.

For every successful run

$$\gamma_{\mathcal{A}} = (q_1, a_1, q_2, \ldots q_{m-1}, a_{m-1}, q_m)$$

of $\mathcal{A}$, where $m \in \mathbb{N}^+$, $q_1, \ldots, q_m \in [\![1, n]\!]$, $a_1, \ldots, a_{m-1} \in \Sigma$, we construct a run

$$\gamma = (p_1, b_1, p_2, \ldots, p_{l-1}, b_{l-1}, p_l)$$

of $\mathcal{A}$ where $l \in \mathbb{N}^+$, $p_1, \ldots, p_l$ is the longest scattered subsequence of $q_1, ..., q_m$ such that $p_i \neq p_j$ for every $i, j \in [\![1, l]\!]$ and $b_1, ..., b_{l-1} \in \Sigma$ are letters such that (p_i, b_i, p_{i+1})

[1]The number 0 denotes the neutral element of an additive operation.

is a transition in $\gamma_\mathcal{A}$ for every $i \in [\![1, l-1]\!]$. The construction is unambiguous and γ is a successful run with nonrepeating states. Similarly as above, we can construct a branchless automaton $\mathcal{A}_\gamma$ and then construct a run

$$\gamma_\mathcal{B} = (p_{\gamma,q_1}, a_1, p_{\gamma,q_3}, \dots, p_{\gamma,q_{m-1}}, a_{m-1}, p_{\gamma,q_m})$$

of $\mathcal{A}_\gamma$ and hence of $\mathcal{B}$.

For a given successful run $\gamma_\mathcal{B} = (p_{\gamma,q_1}, a_1, p_{\gamma,q_2}, \dots, p_{\gamma,q_{m-1}}, a_{m-1}, p_{\gamma,q_m})$ of $\mathcal{B}$ where $m \in \mathbb{N}^+$, γ is a run with nonrepeating states, $q_1, \dots q_m \in [\![1, n]\!]$, $p_{\gamma,q_1}, \dots, p_{\gamma,q_m} \in Q_\gamma$, and $a_1, ..., a_m \in \Sigma$, we construct a run $\gamma_\mathcal{A} = (q_1, a_1, q_2, \dots, q_{m-1}, a_{m-1}, q_m)$ of $\mathcal{A}$.

We see that the construction is a bijection. The equality $\sigma_\mathcal{A}(\gamma_\mathcal{A}) = \sigma_\mathcal{B}(\gamma_\mathcal{B})$ holds obviously by how $\mathcal{A}_\gamma$ is constructed. $\square$

Remark 3.1.5. For deterministic weighted automata, the branchless normal form would require joining the initial states of all branches.

Remark 3.1.6. The normalisation shown above is not applicable in algorithms which require polynomial time, as the normalisation may expand the size of an automaton exponentially. For instance, every automaton in the branchless normal form which realises the same series as the automaton shown in Figure 3.3 has at least 2^n states. Hence, we can construct a series of such automata for every $n \in \mathbb{N}$ to see that the space complexity of normalisation belongs to $O(2^n)$, where n is the number of states of an automaton.

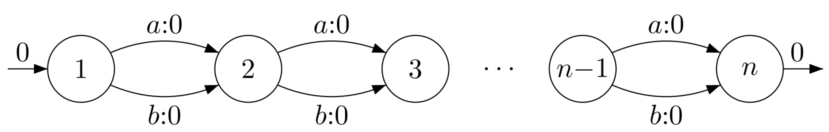

Figure 3.3: An acyclic automaton with loops whose normal form has at least 2^n states.

3.2 Characterisation of Realised Series

In this section, we characterise rational series realised by acyclic automata with loops.

In the unweighted case, the class of languages realised by acyclic automata with loops is equivalent to the level 3/2 of the Straubing-Therien hierarchy [3], also known as the class of alphabetic pattern constraints [4]. The deterministic variant realises the class of R-trivial languages [5].

Theorem 3.2.1. Let S be a semiring, Σ an alphabet and $r \in S\langle\!\langle\Sigma^*\rangle\!\rangle$. There exists an acyclic weighted automaton with loops $\mathcal{A} = (Q, \sigma, \iota, \tau)$ over S and Σ such that $||\mathcal{A}|| = r$

iff r is a finite sum of the series taking the form $v_0 R_1^* v_1 a_1 R_2^* v_2 \dots v_{m-1} a_{m-1} R_m^* v_m$ where $m \in \mathbb{N}^+, v_0, \dots, v_m \in S, a_1, \dots, a_{m-1} \in \Sigma$ and $R_1, \dots, R_m \in S\langle\Sigma\rangle$.

Proof.

- $\Rightarrow$ For a given acyclic weighted automaton $\mathcal{A}$ over S and Σ, we construct an automaton $\mathcal{A}'$ over S and Σ which is in the branchless normal form such that $||\mathcal{A}'|| = ||\mathcal{A}||$. Let $\mathcal{A}'_1, ..., \mathcal{A}'_B$ be branchless automata for some $B \in \mathbb{N}^+$ such that $\mathcal{A}' = \mathcal{A}'_1 + ... + \mathcal{A}'_B$. By Theorem 2.1.1 for every $i \in [\![1, B]\!]$ there exists a series $r_i = v_0 R_1^* v_1 a_1 R_2^* v_2 \dots v_{m-1} a_{m-1} R_m^* v_m$ where $m \in \mathbb{N}^+, v_0, \dots, v_m \in S, a_1, \dots, a_{m-1} \in \Sigma$ and $R_1, \dots, R_m \in S\langle\Sigma\rangle$, such that $r_i = ||\mathcal{A}'_i||$. By Proposition 1.3.9 we have $||\mathcal{A}'|| = \sum_{i=1}^{B} ||\mathcal{A}'_i|| = \sum_{i=1}^{B} r_i$.

- $\Leftarrow$ Let $r = \sum_{i=1}^{B} r_i$ be a given series where $B \in \mathbb{N}^+$ and for every $i \in [\![1, B]\!]$, let $r_i = v_0 R_1^* v_1 a_1 R_2^* v_2 \dots v_{m-1} a_{m-1} R_m^* v_m$ where $m \in \mathbb{N}^+, v_0, \dots, v_m \in S, a_1, \dots, a_{m-1} \in \Sigma$ and $R_1, \dots, R_m \in S\langle\Sigma\rangle$. By Theorem 2.1.1, we construct a branchless automaton $\mathcal{A}_i = (Q_i, \sigma_i, \iota_i, \tau_i)$ over S and Σ for every $i \in [\![1, B]\!]$ such that $||\mathcal{A}_i|| = r_i$. Then, we construct an automaton $\mathcal{A} = (Q, \sigma, \iota, \tau)$ over S and Σ such that $\mathcal{A} = \mathcal{A}_1 + \dots + \mathcal{A}_B$. By Proposition 1.3.9 we have $||\mathcal{A}|| = \sum_{i=1}^{B} ||\mathcal{A}'_i|| = \sum_{i=1}^{B} r_i = r$ and since a disjoint union of acyclic automata with loops is an acyclic automaton with loops, the proof is complete. $\square$

3.3 Hardness Results

The class of acyclic weighted automata with loops is a subclass of general weighted automata and a superclass of branchless weighted automata. Therefore, in proving results for acyclic automata with loops, we can use hardness results proved for branchless automata (the $\exists$-exact problem) and decidability results proved for general weighted automata over tropical semirings summarised in the Tables 1.1 and 1.2 in the first chapter.

For the decision problems considered in [2], we show that if a problem is **PSPACE**-complete for general weighted automata, it is **PSPACE**-complete also for acyclic automata with loops. We prove it using a similar approach as in [2] showing a reduction of the **PSPACE**-complete universality problem for unweighted acyclic automata with loops [14] to problems of universality, upper boundedness, $\exists$-exact... over $\mathbb{N}_{min}$.

Theorem 3.3.1. The universality problem for acyclic weighted automata with loops over $\mathbb{N}_{min}$ is **PSPACE**-complete.

Proof. Deciding the universality problem belongs to **PSPACE** as was shown in [2]. To prove **PSPACE**-hardness, we use a reduction of the **PSPACE**-hard universality problem for unweighted acyclic automata with loops [14] to the universality problem for acyclic weighted automata with loops over $\mathbb{N}_{min}$. For a given unweighted acyclic automaton with loops, we construct an acyclic weighted automaton with loops over $\mathbb{N}_{min}$ which has the same states and transitions. We assign each transition a weight of zero, each initial state an initial weight of zero and each terminal state a terminal weight of zero. All other weights are set to ∞. It is clear that the given acyclic automaton accepts Σ^* iff all words from Σ^* in the constructed weighted automaton have a weight smaller than 1. $\square$

S. Almagor, U. Boker, and O. Kupferman proved **PSPACE**-completeness of the following problems for general weighted automata using a reduction from the **PSPACE**-complete universality problem for unweighted automata [2]. Since their reductions do not use any cycles longer than one and the universality problem for unweighted acyclic automata with loops is also **PSPACE**-complete [14], we may prove the following theorem using the same reductions.

Theorem 3.3.2. The $\exists$-exact problem [2, p. 16], the $\forall$-exact problem [2, p. 16], the upper boundedness problem [2, p. 18], the absolute boundedness problem [2, p. 18] and the equality problem of deterministic and nondeterministic automata [2, p. 17] for acyclic weighted automata with loops over $\mathbb{N}_{min}$ are **PSPACE**-complete.

The $\exists$-exact problem for deterministic acyclic weighted automata with loops is **NP**-complete, as was proved in Corollary 2.5.3.

3.4 Undecidable Problems

From the previous section, we see that every problem which is decidable for general automata over tropical semirings belongs to one of the following complexity classes for acyclic automata with loops: **PSPACE**-complete, **NP**-complete or **P**. Therefore the only left open problems are those, that are undecidable for general weighted automata, e.g., the upper boundedness problem over $\mathbb{Z}_{min}$ or the universality problem over $\mathbb{Z}_{min}$.

Proving that these problems are undecidable would require a reduction which is, in some way, stronger than the reductions used for proving the undecidability of the problems for general weighted automata. The reduction demonstrated by S. Almagor, U. Boker, and O. Kupferman is not applicable for acyclic automata with loops, since it uses cycles longer than one [2]. It is unknown, whether we could modify their reduction,

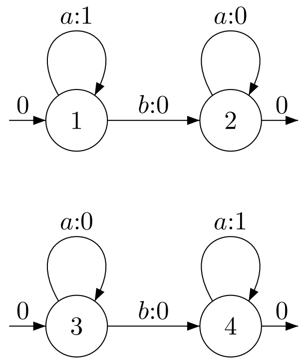

Figure 3.4: An acyclic automaton with loops $\mathcal{A}$ for which $\{(||A||, a^k ba^k), \mid k \in \mathbb{N}\}$ is unbounded.

the Krob's reduction [12], or the Colcombet's reduction [7] to prove undecidability of any problem for acyclic weighted automata with loops.

Despite the unknown result of decidability of these problems, we analysed the upper boundedness problem and formulated the following conjecture.

Conjecture 3.4.1. Let Σ be an alphabet. For a given acyclic weighted automaton with loops $\mathcal{A} = (Q, \sigma, \iota, \tau)$ over $\mathbb{Z}_{min}$ and Σ, the series $||\mathcal{A}||$ is not upper bounded iff there exist $m \in \mathbb{N}$ and $a_1, p_1, \ldots, p_m, a_{m+1} \in \Sigma^*$, for which there does not exist a boundary $B \in \mathbb{N}$ such that $(||A||, a_1 p_1^k \ldots p_{m-1}^k a_m) < B$ for every $k \in \mathbb{N}$.

This conjecture is supported by the fact that for all automata previously described in this thesis which realise series that are not upper bounded, we can find words a_1, p_1, a_2 over considered alphabet for which the set of coefficients $(||\mathcal{A}||, a_1 p_1^* a_2)$ is not upper bounded. However, we can provide an example of the automaton shown in Figure 3.4, which realises unbounded rational series and for which there do not exist words a_1, p_1, a_2 such that $(||\mathcal{A}||, a_1 p_1^* a_2)$ is not upper bounded. Still, we can find words $a_1 = \varepsilon, p_1 = a, a_2 = b, p_2 = a, a_3 = \varepsilon$ such that the set of coefficients $\{(||A||, a_1 p_1^k a_2 p_2^k a_3) \mid k \in \mathbb{N}\}$ is not upper bounded.

If the proposed conjecture is proved true, we could reduce the problem to finding words $a_1, p_1, \ldots, p_m, a_m$ for which the set of coefficients $\{(||A||, a_1 p_1^k \ldots p_{m-1}^k a_m) \mid k \in \mathbb{N}\}$ is not upper bounded. We leave proving or disproving the proposed conjecture for further research.

Conclusion

In this thesis, we examined the decidability of selected problems for acyclic weighted automata with loops and for homogeneous branchless weighted automata over tropical semirings. In the first chapter, we provided the necessary definitions and explained the fundamental concepts and problems we were about to consider.

The second chapter focused on branchless automata. First, we defined the class of automata, proved a normal form theorem for this class and characterised the rational series realised by branchless automata. Then we focused on three decision problems.

For the first problem, upper boundedness, we proved that its deciding belongs to **P** for homogeneous branchless automata with loops over the tropical semiring of natural numbers $\mathbb{N}_{min}$. We first observed that the series realised by an automaton is not upper bounded if there exists a loop with positive weight on a letter different than the common label a of all sequel transitions. Then we proved that the upper boundedness is strongly related to the property of the so-called supersession of its states, i.e., the series realised by an automaton is upper bounded if and only if all states with a positive loop on the letter a are superseded by some state with a zero loop on the letter a. This result implies that the absolute boundedness problem is decidable in polynomial time for homogeneous branchless automata over $\mathbb{N}_{min}$ and $\mathbb{Z}_{min}$.

The universality problem over the *support* of the realised series was proved to belong to **co**-**NP** for homogeneous branchless automata over $\mathbb{N}_{min}$, because for upper bounded homogeneous branchless automata, we only need to consider a finite language of words, which we can search in **co**-**NP** time. The same result was then obtained for the $\forall$-exact problem for homogeneous branchless automata over $\mathbb{N}_{min}$ and $\mathbb{Z}_{min}$.

The third problem, $\exists$-exact, is known to be **NP**-hard for acyclic weighted automata with loops [2]. We have presented a reduction which proves **NP**-hardness for (both deterministic and nondeterministic) *one state* weighted automata over $\mathbb{N}_{min}$ and $\mathbb{Z}_{min}$.

In the third chapter, we considered acyclic weighted automata with loops, first defining the class of automata and characterising the rational series realised by them. We also presented a normal form of acyclic automata with loops – branchless normal form – in which the automaton takes the form of a disjoint union of multiple several branchless automata.

For the class of nondeterministic acyclic automata with loops over $\mathbb{N}_{min}$, we demonstrated that most problems remain **PSPACE**-hard, showing a reduction from a **PSPACE**-hard problem for unweighted acyclic automata with loops.

Several problems still remain open, e.g., the decidability status of some decision problems for nondeterministic acyclic automata with loops over $\mathbb{Z}_{min}$. For the upper boundedness problem, we proposed a conjecture which, if proved true, would possibly enable deciding upper boundedness for acyclic automata with loops over semiring $\mathbb{Z}_{min}$. It also might be interesting to study the decision problems for branchless automata which are not homogeneous.

Bibliography

[1] Shaull Almagor, Udi Boker, and Orna Kupferman. What's decidable about weighted automata? In *ATVA 2011*, volume 6996, pages 482–491, 2011.

[2] Shaull Almagor, Udi Boker, and Orna Kupferman. What's decidable about weighted automata? *Information and Computation*, 282, 2022. Article 104651.

[3] Mustapha Arfi. Opérations polynomiales et hiérarchies de concaténation. *Theoretical Computer Science*, 91(1):71–84, 1991.

[4] Ahmed Bouajjani, Anca Muscholl, and Tayssir Touilim. Permutation rewriting and algorithmic verification. In *LICS 2001*, pages 399–408. IEEE, 2001.

[5] Janusz A. Brzozowski and Faith E Fich. Languages of R-trivial monoids. *Journal of Computer and System Sciences*, 20(1):32–49, 1980.

[6] Manfred Droste and Werner Kuich. Semirings and formal power series. In Manfred Droste, Werner Kuich, and Heiko Vogler, editors, *Handbook of Weighted Automata*, chapter 1, pages 3–28. Springer, 2009.

[7] Manfred Droste and Dietrich Kuske. Weighted automata. In Jean-Éric Pin, editor, *Handbook of Automata Theory, Vol. 1*, chapter 4, pages 113–150. European Mathematical Society, 2021.

[8] Michael R Garey and David S Johnson. *Computers and intractability*, volume 174. W.H.Freeman, 1979.

[9] K. A. Hansen. Simple reduction to unbounded knapsack? Theoretical Computer Science Stack Exchange. Retrieved May 18, 2022, from https://cstheory.stackexchange.com/q/1002.

[10] Kosaburo Hashiguchi. Limitedness theorem on finite automata with distance functions. *Journal of computer and system sciences*, 24(2):233–244, 1982.

[11] Ondřej Klíma and Libor Polák. Alternative automata characterization of piecewise testable languages. In *DLT 2013*, pages 289–300. Springer, 2013.

[12] Daniel 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.

[13] Markus Krötzsch and Tomáš Masopust. Partially ordered automata and piecewise testability. *Logical Methods in Computer Science*, 17, 2021.

[14] Markus Krötzsch, Tomáš Masopust, and Michaël Thomazo. Complexity of universality and related problems for partially ordered NFAs. *Information and Computation*, 255:177–192, 2017.

[15] Mehryar Mohri. Weighted automata algorithms. In Manfred Droste, Werner Kuich, and Heiko Vogler, editors, *Handbook of Weighted Automata*, chapter 6, pages 213–256. Springer, 2009.

[16] Jean-Éric Pin. Mathematical foundations of automata theory. *Lecture notes LIAFA, Université Paris*, 7, 2010.

[17] Jacques Sakarovitch. *Elements of automata theory.* Cambridge University Press, 2009.

[18] Zoltán Ésik and Werner Kuich. Finite automata. In Manfred Droste, Werner Kuich, and Heiko Vogler, editors, *Handbook of Weighted Automata*, chapter 3, pages 69–104. Springer, 2009.