Diplomarbeit

Decidability of the Twins Property for Weighted Tree Automata over Extremal Semirings

Anja Fischer

24. März 2012

Technische Universität Dresden
Fakultät Informatik
Institut für Theoretische Informatik
Professur Theorie der Programmierung

Betreuender Hochschullehrer: Prof. Dr.-Ing. habil. Heiko Vogler
Betreuender Mitarbeiter: Dipl.-Inf. Matthias Büchse

Anja Fischer

Erklärung

Hiermit erkläre ich, dass ich diese Arbeit selbstständig erstellt und keine anderen als die angegebenen Hilfsmittel benutzt habe.

Dresden, den 24. März 2012

Anja Fischer

Danksagung

An dieser Stelle möchte ich mich bei all denen bedanken, die mich beim Anfertigen dieser Arbeit tatkräftig unterstützt und motiviert haben, und ohne die die vergangenen fünfeinhalb Jahre nur halb so schön gewesen wären.

Zunächst zu nennen wären da Matthias Büchse für viele hilfreiche Tipps und die Betreuung während der letzten Monate sowie Prof. Vogler für seine Unterstützung bei der kurzfristigen Umgestaltung meines weiteren Studienverlaufes als sich plötzlich unvorhergesehene Hindernisse wie der Wegfall fest eingeplanter Lehrveranstaltungen auftaten.

Zudem gilt mein Dank meiner Familie, insbesondere meinen Eltern, die mir ein sorgenfreies Studium ermöglicht haben, mich motiviert haben und mich bei allen meinen Projekten wie meinem Auslandssemester unterstützt haben.

Nicht vergessen möchte ich auch die Korrekturleser dieser Arbeit. Ich kann mir gut vorstellen, dass ihr bestimmt auch Besseres zu tun gehabt hättet... ;-)

Und schließlich geht mein Dank an meine Freunde und Kommilitonen, die das Studium hier in Dresden zu einer unvergesslichen Zeit gemacht haben!

Aufgabenstellung für die Diplomarbeit

"Decidability of the Twins Property
for Weighted Tree Automata over Extremal Semirings"

Technische Universität Dresden
Fakultät Informatik

Studentin: Anja Fischer
Studiengang: Informatik
Matrikelnummer: 3295958

Die Klasse der bottom-up-deterministisch erkennbaren gewichteten Baumsprachen [1] ist *echt* in der Klasse der erkennbaren gewichteten Baumsprachen enthalten [BV03]. Folglich ist eine Determinisierung gewichteter Baumautomaten nicht uneingeschränkt möglich. Dies drückt sich darin aus, dass die bekannten Verfahren nicht in jedem Fall terminieren.

Es wurden bislang zwei hinreichende Kriterien für die Terminierung angegeben: (a) der Halbring ist lokal endlich [Bor04] und (b) der Halbring ist extremal und der gewichtete Baumautomat erfüllt die Twins-Eigenschaft [BMV10]. Dabei ist bislang offen, ob die Twins-Eigenschaft entscheidbar ist. Ein aktuelles Resultat von Kirsten [Kir10] für gewichtete String-Automaten legt nun zumindest nahe, dass dies für gewichtete Baumautomaten über einem extremalen Halbring der Fall ist.

Die Aufgabe dieser Diplomarbeit besteht darin, den Beweis von Kirsten in die hier angesprochene Baumwelt zu übertragen. Falls sich dies als unmöglich herausstellt, soll dies mit Argumenten glaubhaft belegt werden und nach Möglichkeit eine Alternative erschlossen werden. Optional kann die Studentin (i) einen Test auf Twins-Eigenschaft in HASKELL implementieren, passend zum System VANDA des Lehrstuhls, oder (ii) das Resultat oder den Beweis aufwerten, zum Beispiel durch eine besonders elegante Darstellung.

Die Arbeit muss den üblichen Standards wie folgt genügen. Die Arbeit muss in sich abgeschlossen sein und alle nötigen Definitionen und Referenzen enthalten. Die Struktur der Arbeit muss klar erkenntlich sein, und der Leser soll gut durch die Arbeit geführt werden. Die Darstellung aller Begriffe und Verfahren soll mathematisch formal fundiert sein. Für jeden wichtigen Begriff sollen Beispiele angegeben werden, ebenso für die Abläufe der beschriebenen Verfahren. Wo es angemessen ist, sollten Illustrationen die Darstellung vervollständigen. Schließlich sollen alle Lemmata und Sätze möglichst lückenlos bewiesen werden. Die Beweise sollen leicht nachvollziehbar dokumentiert sein. Im Falle einer Implementation soll eine ausführliche Dokumentation erfolgen, die sich angemessen auf den Quelltext und die Diplomarbeit verteilt. Dabei muss die Funktionsfähigkeit des Programms glaubhaft gemacht werden.

Literatur

[BMV10] Matthias Büchse, Jonathan May, and Heiko Vogler. Determinization of weighted tree automata using factorizations. *Journal of Automata, Languages and Combinatorics*, 15(3/4), 2010.

[Bor04] Björn Borchardt. A pumping lemma and decidability problems for recognizable tree series. *Acta Cybern.*, 16:509–544, September 2004.

[BV03] Björn Borchardt and Heiko Vogler. Determinization of finite state weighted tree automata. *J. Autom. Lang. Combin*, 8, 2003.

[Kir10] Daniel Kirsten. Decidability, undecidability, and pspace-completeness of the twins property in the tropical semiring. Manuscript, 2010.

[1] In der Literatur werden gewichtete Baumsprachen auch als Baumreihen bezeichnet.

Contents

List of Figures

1 Introduction

Machine Translation aims to provide the means to automatically translate one natural language, like English or German, into another.

In Statistical Machine Translation (SMT), translation rules are usually extracted from a set of example translations, a parallel corpus, and accumulated within a grammar. The weight of each rule is estimated using statistical methods. This way, the weight of an object derived from the newly created grammar matches approximately its likelihood within the corpus.

However, grammars often prove to be *ambiguous*, i. e., they admit several possible derivations of the same object. Consider the following sentence: *He gave her cat food.* Did he give cat food to someone or did he simply feed someone's cat? In the first interpretation, "her" is the object of the sentence, whereas in the second case "her cat" is the object. Thus, the different meanings are hidden within the structure of the sentence's derivations, invisible to the reader of that sentence. This structure can be represented by a derivation tree. In contrast to phrase-based machine translation, where there is no information available other than the example sentences themselves, the syntax-based approach makes great use of the information that derivation trees provide.

Given an ambiguous grammar, the likelihood of an object is estimated by cumulating all its derivations. This cumulation, however, is computationally intractable: it has been shown that finding the best string that is derivable from a given grammar is NP-hard [Sim96, CdlH00]. On the other hand, finding a best derivation is easier: it can be done in polynomial time [Epp98, HC05].

This fact led to the idea of approximating an object's likelihood by its best derivation. Since derivations can be represented by trees, the application of tree automata [GS84] seems obvious. These devices work with trees rather than strings, and their weighted counterparts, *weighted tree automata* (wta), assign a weight to every tree in the same way as a weight has been assigned to every derivation. Much like grammars, wta can be ambiguous or nondeterministic, i. e., they admit several trees with the same yield. Returning to the original idea of describing an object's likelihood by its best derivation, determinization of wta seems to be a reasonable solution to the ambiguity problem: once we have a deterministic wta, the best derivation of an object is its only derivation.

However, determinizing a wta is not always possible, as the class of deterministic wta is strictly less powerful than the class of nondeterministic wta. This problem led to research on sufficient criteria that allow determinization. Mohri [Moh97] was one of the first to work on that problem. He was able to prove that weighted finite string automata (wfa), finite-state machines that work with strings rather than trees, over the tropical semiring are determinizable if they have the *twins property*, a notion described earlier in [Cho77]. This result has been generalized to wfa over arbitrary semirings using a mathematical concept called *factorization* [KM05].

Borchardt and Vogler were the first to describe determinization of wta [BV03]. They limited their algorithm to wta over locally finite semifields before the approach was extended to locally finite semirings in [Bor04]. In [BMV10] another determinization algorithm for weighted tree automata is presented, subsuming the results of the aforementioned papers.

Despite its importance for determinization algorithms the decidability of the twins property itself remained open. The first result in this area was presented by Allauzen and Mohri [AM03] who show the decidability of the twins property for the class of cycle unambiguous wfa. Moreover, they proved that deciding the twins property in that class can be done in polynomial time. However, the question as to whether the twins property is also decidable for other classes of weighted automata remained. Recent work on this topic by Kirsten [Kir12] proves the decidability of the twins property for wfa over the tropical semiring and suggests a possible generalization to trees. Kirsten also showed that deciding the twins property was PSPACE-complete. We pursue Kirsten's suggestion and aim to prove the decidability of the twins property for wta over extremal semirings.

1.1 Outline

As a preparation, Chapter 2 recalls important concepts such as ranked trees, semirings and factorizations and presents some smaller examples.

Chapter 3 gives the definition of weighted tree automata, their semantics and the twins property. Again, examples are shown to facilitate comprehension and to improve intuition.

Our main theorem, its proof, and two decision algorithms are presented in Chapter 4. Finally, Chapter 5 concludes with a short summary of our work and proposes possible future work emanating from this thesis.

2 Preliminaries

2.1 General Notions on Ranked Trees

By $\mathbb{N}$ we denote the set of natural numbers, i. e., non-negative integers.

Definition 2.1 The *star* $\mathbb{N}^*$ is defined as the set $\mathbb{N}^* = \bigcup_{i \in \mathbb{N}} \mathbb{N}^i$, where $\mathbb{N}^i$ is defined inductively for every $i \in \mathbb{N}$:

(i) $\mathbb{N}^0 = \{\varepsilon\}$ and

(ii) $\mathbb{N}^{i+1} = \{nm \mid n \in \mathbb{N}, m \in \mathbb{N}^i\}$, where nm denotes the concatenation of n and m. $\square$

Definition 2.2 A *ranked alphabet* is a pair (Σ, rk) where Σ is an alphabet, i. e., a finite nonempty set, and rk is a mapping $rk\colon \Sigma \to \mathbb{N}$ that assigns a natural *arity* to every symbol $\sigma \in \Sigma$. $\square$

Henceforth, we identify (Σ, rk) with Σ. Furthermore, we define for every $k \in \mathbb{N}$ the set $\Sigma^{(k)} = \{\sigma \in \Sigma \mid rk(\sigma) = k\}$ containing all symbols of arity k. A symbol $\sigma \in \Sigma^{(k)}$ is also abbreviated with $\sigma^{(k)}$.

Definition 2.3 The set $T_\Sigma(H)$ of *ranked trees over* Σ *indexed by* H is defined as the smallest set T such that:

(i) $H \subseteq T$

(ii) $\sigma(\xi_1, \ldots, \xi_k) \in T$ for every $k \in \mathbb{N}$, $\sigma \in \Sigma^{(k)}$, and $\xi_1, \ldots, \xi_k \in T$. $\square$

Moreover, we write T_Σ instead of $T_\Sigma(\emptyset)$.

Definition 2.4 For every $\xi \in T_\Sigma(H)$, we define the set $\mathrm{pos}(\xi) \subseteq \mathbb{N}^*$ of *positions of* ξ as follows.

(i) if $\xi \in H$, then $\mathrm{pos}(\xi) = \{\varepsilon\}$, and

(ii) if $\xi = \sigma(\xi_1, \ldots, \xi_k)$, then $\mathrm{pos}(\xi) = \{\varepsilon\} \cup \{iw \mid i \in \{1, \ldots, k\}, w \in \mathrm{pos}(\xi_i)\}$. $\square$

Definition 2.5 For every $\xi \in T_\Sigma(H)$, we define the mapping $\mathrm{ht}\colon T_\Sigma(H) \to \mathbb{N}$ that maps every tree ξ to its *height*, i. e., the length of a longest position $w \in \mathrm{pos}(\xi)$:

(i) if $\xi \in H$ or $\xi \in \Sigma^{(0)}$, then $\mathrm{ht}(\xi) = 0$, and

(ii) if $\xi = \sigma(\xi_1, \ldots, \xi_k)$, then $\mathrm{ht}(\xi) = 1 + max\{\mathrm{ht}(\xi_\mathrm{i}) \mid i \in \{1, \ldots, k\}\}$. $\square$

Definition 2.6 Let $\xi, \xi' \in T_\Sigma(H)$ and $w \in \mathrm{pos}(\xi)$. We define the following mappings: The *label* of ξ at position w:

$$\xi(w) = \begin{cases} \xi & \text{if } \xi \in H, w = \varepsilon \\ \sigma & \text{if } \xi = \sigma(\xi_1, \dots, \xi_k), k \geq 0, \sigma \in \Sigma^{(k)}, w = \varepsilon \\ \xi_i(w') & \text{if } \xi = \sigma(\xi_1, \dots, \xi_i, \dots, \xi_k), k \geq 0, \sigma \in \Sigma^{(k)}, \\ & \quad i \in \{1, \dots, k\}, \text{ and } w = iw' \end{cases}$$

The *subtree* of ξ rooted at position w:

$$\xi|_w = \begin{cases} \xi & \text{if } w = \varepsilon \\ \xi_i|_{w'} & \text{if } \xi = \sigma(\xi_1, \dots, \xi_i, \dots, \xi_k), k \geq 0, \sigma \in \Sigma^{(k)}, \\ & \quad i \in \{1, \dots, k\}, \text{ and } w = iw' \end{cases}$$

The *substitution* of the subtree of ξ at position w with ξ':

$$\xi[\xi']_w = \begin{cases} \xi' & \text{if } w = \varepsilon \\ \sigma(\xi_1, \dots, \xi_i[\xi']_{w'}, \dots, \xi_k) & \text{if } \xi = \sigma(\xi_1, \dots, \xi_i, \dots, \xi_k), k \geq 0, \sigma \in \Sigma^{(k)}, \\ & \quad i \in \{1, \dots, k\}, \text{ and } w = iw' \end{cases}$$

□

Definition 2.7 A Σ-*context* is a tree $\zeta \in T_\Sigma(\{z\})$ that contains exactly one occurrence of the special symbol z. The set of all Σ-contexts is denoted by C_Σ. □

Definition 2.8 Let $\xi \in T_\Sigma \cup C_\Sigma$ and $\zeta \in C_\Sigma$. The *concatenation of* ξ *and* ζ, denoted by $\xi \cdot \zeta$, is obtained from replacing the leaf z in ζ by ξ. □

Concatenation is an associative operation, i. e., $(\xi \cdot \zeta_1) \cdot \zeta_2 = \xi \cdot (\zeta_1 \cdot \zeta_2)$ for every $\zeta_1, \zeta_2 \in C_\Sigma$ and $\xi \in T_\Sigma$. Let $\zeta \in C_\Sigma$. If $\xi \in T_\Sigma$, then so is $\xi \cdot \zeta$. Otherwise, if $\xi \in C_\Sigma$, then the same applies to $\xi \cdot \zeta$.

2.2 Semirings

Definition 2.9 A *semiring* [Gol99, HW98] is a tuple $\mathcal{S} = (S, +, \cdot, 0, 1)$ where S is an arbitrary set called *carrier set*, $+$ and $\cdot$ are binary operations called *addition* and *multiplication*, and 0 and 1 are elements of S such that:

- $(S, +, 0)$ is a commutative monoid, i. e., $+$ is associative, commutative and 0 is the neutral element satisfying $a + 0 = a = 0 + a$ for all $a \in S$,
- $(S, \cdot, 1)$ is a monoid,
- multiplication distributes over addition from both sides, i. e.,
  $\forall a, b, c \in S\colon a \cdot (b + c) = a \cdot b + a \cdot c$ and $(a + b) \cdot c = a \cdot c + b \cdot c$,

- 0 is absorbing with respect to multiplication, i. e.,
  $\forall a \in S\colon 0 \cdot a = 0 = a \cdot 0$. □

Let $\mathcal{S} = (S, +, \cdot, 0, 1)$. Henceforth, we will identify $\mathcal{S}$ with its carrier set S. We call S *commutative* if the multiplication is commutative and a *semifield* if it is commutative and for every $s \in S \setminus \{0\}$ there is an element $s^{-1} \in S$ such that $s^{-1} \cdot s = 1$. Furthermore, S is *zero-sum free* if $a + b = 0$ implies $a = b = 0$; it is *zero-divisor free* if $a \cdot b = 0$ implies $a = 0$ or $b = 0$. Moreover, S is *cancellative* if $a \cdot b = a \cdot c$ implies $b = c$. We consider S to be *extremal* [Mah84] if $a + b \in \{a, b\}$ for every $a, b \in S$. We note that every extremal semiring is also zero-sum free and every semifield is zero-divisor free.

Example 2.10 We present four examples of semirings.

- The *Boolean semiring* $\mathbb{B} = (\{0, 1\}, \vee, \wedge, 0, 1)$ with disjunction and conjunction is extremal and a semifield, and therefore zero-sum free and zero-divisor free.
- The *formal language semiring* $(\mathcal{P}(\Sigma^*), \cup, \cdot, \emptyset, \{\varepsilon\})$ over an alphabet Σ with union and language concatenation is neither commutative nor extremal. However, it is zero-divisor free and zero-sum free.
- The *tropical semiring* $(\mathbb{R} \cup \{\infty\}, \min, +, \infty, 0)$ with minimum and conventional addition is an extremal semifield.
- The *Viterbi semiring* $([0, 1], \max, \cdot, 0, 1)$ is commutative, extremal, and zero-divisor free. It is, however, not a semifield. □

The set S^Q contains all mappings $u\colon Q \to S$. Thus, S^Q can be viewed as the set containing all Q-vectors over S. Instead of $u(q)$ we denote the q-component of every $u \in S^Q$ by u_q. The Q-vector mapping every $q \in Q$ to 0 is denoted by $\tilde{0}$.

Definition 2.11 For every $q \in Q$ we define a corresponding *e-vector*, denoted by $e_q \in S^Q$, that maps only its q-component to the semiring one while all other entries are mapped to the semiring zero, i. e., $(e_q)_q = 1$ and $(e_q)_p = 0$ for every $p \in Q$ with $p \neq q$. □

2.3 Factorization

Factorizations provide the means to extract a common factor from all components of a given vector and are crucial to our later proofs. Throughout this thesis we use the notion of a factorization as specified in [KM05].

Definition 2.12 Let Q be a nonempty, finite set and S an arbitrary semiring. A pair (f, g) is called a *factorization of dimension* Q if $f\colon S^Q \setminus \{\tilde{0}\} \to S^Q$, $g\colon S^Q \setminus \{\tilde{0}\} \to S$, and $u = g(u) \cdot f(u)$ for every $u \in S^Q \setminus \{\tilde{0}\}$. A factorization (f, g) is called *maximal* if for every $u \in S^Q$ and $a \in S$, we have that $a \cdot u \neq \tilde{0}$ implies $f(a \cdot u) = f(u)$. □

As shown in [BMV10, Lemma 4.4], a maximal factorization only exists if S is zero-divisor free or $|Q| = 1$.

Example 2.13 Let Q be a nonempty, finite set and (f, g) a factorization of dimension Q. We give three examples of factorizations over various semirings:

- Let S be an arbitrary semiring and (f, g) the *trivial factorization*, i. e., $g(u) = 1$ and $f(u) = u$ for every $u \in S^Q \setminus \{\tilde{0}\}$. This factorization (f, g) is not maximal in general.
- Let S be the tropical semiring and (f, g) such that $g(u) = \min\{u_q \mid q \in Q\}$ and $f(u) = -g(u) + u$ for every $u \in S^Q \setminus \{\tilde{0}\}$. Thus, (f, g) is maximal: $f(a + u) = -\max\{a + u_q \mid q \in Q\} + a + u = -a - g(u) + a + u = f(u)$.
- Let $S = (\mathbb{R}^{\geq 0}, +, \cdot, 0, 1)$ be the semifield of non-negative reals. Then $g(u) = \sum_{q \in Q} u_q$ and $f(u) = \frac{1}{g(u)} \cdot u$ constitute a factorization, which is also maximal. We note that this construction of a maximal factorization applies to every semifield. $\square$

3 Weighted Tree Automata

A weighted tree automaton (wta) [ÉK03] is a finite-state machine that represents a *weighted tree language*, i. e., a mapping $\varphi\colon T_\Sigma \to S$. It reads and processes an input tree and assigns a weight based on weighted *transitions*, which constitute the core of the wta.

Definition 3.1 A *weighted tree automaton* is a tuple $\mathcal{A} = (Q, \Sigma, S, \delta, \nu)$ such that Q is a nonempty, finite set of *states*, Σ is a ranked alphabet, S is a semiring, δ is a *transition mapping*, mapping *transitions* $(q_1 \ldots q_k, \sigma, q)$ into S, with $q_1, \ldots, q_k, q \in Q$ and $\sigma \in \Sigma^{(k)}$, and $\nu \in S^Q$ is a vector that maps every state to its *root weight*. □

Example 3.2 Let $\mathcal{A} = (Q, \Sigma, \mathcal{S}, \delta, \nu)$ a wta with $Q = \{q_1, q_2\}$, $\Sigma = \{\alpha^{(0)}, \gamma^{(1)}, \sigma^{(2)}\}$, a semiring $\mathcal{S} = (\mathcal{P}(\mathbb{N}^*), \cup, \circ^R, \emptyset, \{\varepsilon\})$ with union and $\circ^R$ such that $U \circ^R V = \{vu \mid u \in U, v \in V\}$ for every $U, V \in \mathcal{P}(\mathbb{N}^*)$, $\nu = (\{\varepsilon\}, \emptyset)$, and δ as shown by the directed functional hypergraph in Figure 1.

Every node (depicted as a circle) represents a state, whereas transistions are represented by hyperedges (depicted as boxes with arbitrarily many incoming and exactly one outgoing arc). Note that incoming arcs of a hyperedge are read counter-clockwise, beginning at the outgoing arc. The root weight $\nu_{q_1} = \{\varepsilon\}$ is represented by an additional outgoing arc originating from q_1. Every transition that is not shown is mapped to $\emptyset$. □

3.1 Semantics

In the literature we find two different yet equivalent definitions of wta semantics: initial-algebra semantics [GTWW77] and run semantics [FV09, Sec. 3.2]. Throughout this thesis we will only use the latter setup, as it constitutes the basis of our proofs.

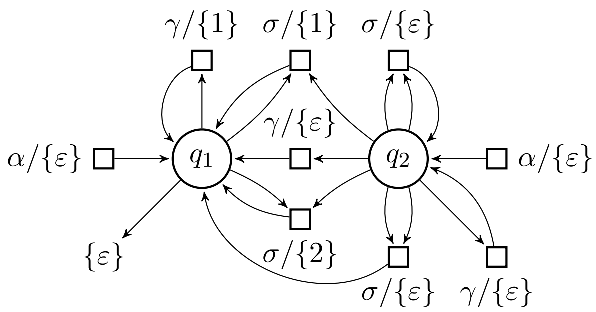

Figure 1: Hypergraph representation of wta $\mathcal{A}$.

In the *run semantics* every node of a given tree is decorated with a state. Such a decoration is henceforth called a *run*. A node's label, its state and its successor states comprise a transition. The *weight of a run* is the product of the weigts of all these transitions (under δ and calculated in S).

The *weight of a tree* is then the sum of the weights of all runs on that tree, again calculated in S. Since a wta represents a weighted tree language, every tree has to be mapped to a certain weight. Those weights are obtained by summing up the componentwise product of a tree's weight and the wta's root weight vector ν.

Throughout this thesis we need to compose and decompose trees and contexts as well as their weights. Thus, we have to consider trees which are indexed by semiring elements as well as Q-vectors over S.

Let H be an index set, $\xi \in T_\Sigma(H)$, $\mathcal{A}$ a wta, and $q \in Q$.

Definition 3.3 The *set of runs on ξ that end in state q at the root of ξ* is defined as follows:

$$R^q(\xi)_\mathcal{A} = \{(\xi, \kappa) \mid \kappa\colon \mathrm{pos}(\xi) \to Q, \kappa(\varepsilon) = q\}. \qquad \square$$

We identify the pair (ξ, κ) with κ and omit the subscript $\mathcal{A}$ if the chosen wta is clear from context.

Definition 3.4 The *set of all runs* regardless of their final state is

$$R(\xi) = \bigcup_{q \in Q} R^q(\xi). \qquad \square$$

Definition 3.5 Let $\kappa \in R^q(\xi)$ and $w \in \mathrm{pos}(\xi)$. We define the following notions:

- $\kappa|_w \in R^{\kappa(w)}(\xi|_w)$ such that for every $w' \in \mathrm{pos}(\xi|_w)\colon \kappa|_w(w') = \kappa(ww')$,
- $\kappa[\kappa']_w \in R^q(\xi[\xi']_w)$ for every $\xi' \in T_\Sigma(H)$ and $\kappa' \in R^{\kappa(w)}(\xi')$ such that
  (i) $\kappa[\kappa']_w$ coincides with κ' on ξ', i. e., $\forall w' \in \mathrm{pos}(\xi')\colon \kappa[\kappa']_w(ww') = \kappa'(w')$, and
  (ii) $\kappa[\kappa']_w$ coincides with κ on $\xi[z]_w$, i. e., $\forall w'' \in \mathrm{pos}(\xi[z]_w)\colon \kappa[\kappa']_w(w'') = \kappa(w'')$,
- $\kappa \cdot \kappa' \in R^{q'}(\xi \cdot \zeta)$ for every $q' \in Q$, $\zeta \in C_\Sigma$, and $\kappa' \in R^{q'}(\zeta)$ mapping the position of node z to state q such that $\kappa \cdot \kappa'$ coincides with κ on ξ and $\kappa \cdot \kappa'$ coincides with κ' on ζ. $\square$

Moreover, we use the following abbreviations: $\kappa[z]_w$ denotes $\kappa[\kappa']_w$ where κ' is the only element of $R^{\kappa(w)}(z)$, and for every $s \in S$ we write $s \cdot \kappa$ to denote the run on $s \cdot \xi$ which coincides with κ. Figure 2 illustrates the two runs $\kappa|_w$ and $\kappa[z]_w$ that are obtained from a given $\kappa \in R(\xi)$.

Definition 3.6 Let $\xi \in T_\Sigma(S \cup S^Q)$ and $\kappa \in R(\xi)$. The *weight of a run* κ, denoted by $\langle\kappa\rangle_\mathcal{A} \in S$, is:

(i) if $\xi \in S$, then $\langle\kappa\rangle_\mathcal{A} = \xi$,

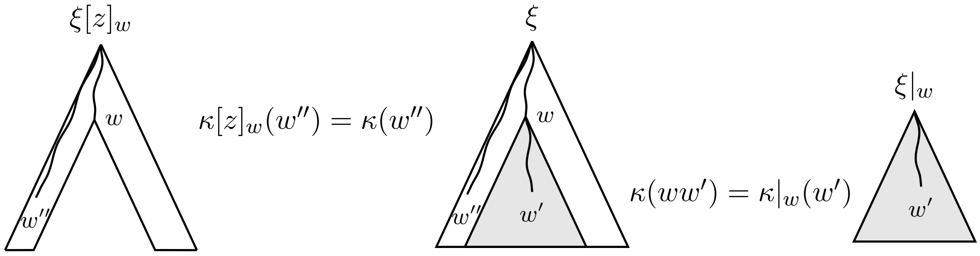

Figure 2: Runs $\kappa[z]_w$ (left-hand side) and $\kappa|_w$ (right-hand side) obtained from a run $\kappa \in R(\xi)$.

(ii) if $\xi \in S^Q$, then $\langle\kappa\rangle_{\mathcal{A}} = \xi_{\kappa(\varepsilon)}$, and

(iii) if $\xi = \sigma(\xi_1, \dots, \xi_k)$ where $\xi_1, \dots, \xi_k \in T_\Sigma(S \cup S^Q)$, $\sigma \in \Sigma^{(k)}$, then we have $\langle\kappa\rangle_{\mathcal{A}} = \langle\kappa|_1\rangle_{\mathcal{A}} \cdot \ldots \cdot \langle\kappa|_k\rangle_{\mathcal{A}} \cdot \delta(\kappa(1)\dots\kappa(k), \sigma, \kappa(\varepsilon))$, calculated in S. $\square$

Again, we omit the subscript $\mathcal{A}$ if the wta is clear from context.

Definition 3.7 We define the mapping $[\![.]\!]_{\mathcal{A}} \colon T_\Sigma(S^Q) \to S^Q$ such that

$$[\![\xi]\!]_{\mathcal{A}}(q) = \sum_{\kappa \in R^q(\xi)} \langle\kappa\rangle. \qquad \square$$

Once more, if $\mathcal{A}$ is clear from context or irrelevant, we will often omit the subscript.

If we have a factorization (f, g), we will write $f[\![\xi]\!]$ instead of $f([\![\xi]\!])$. Moreover, we will often use the following observations. Their proofs are listed in Appendix A.

Observation 3.8 *Let $\xi \in T_\Sigma(S \cup S^Q)$, $q \in Q$, and $\kappa \in R^q(\xi)$. Then we have for every $\zeta \in C_\Sigma$, $q' \in Q$, and $\kappa' \in R^{q'}(\zeta)$ where κ' maps the position of node z to q: $\langle\kappa \cdot \kappa'\rangle = \langle\langle\kappa\rangle \cdot \kappa'\rangle$.*

Observation 3.9 *Let $\xi \in T_\Sigma(S \cup S^Q)$ and $\zeta \in C_\Sigma$. Then, we have $[\![\xi \cdot \zeta]\!] = [\![[\![\xi]\!] \cdot \zeta]\!]$.*

Definition 3.10 The weighted tree language *run-recognized* by a wta $\mathcal{A}$ is the mapping $\varphi_{\mathcal{A}} \colon T_\Sigma \to S$ such that for every $\xi \in T_\Sigma$ we have

$$\varphi_{\mathcal{A}}(\xi) = \sum_{q \in Q} [\![\xi]\!]_q \cdot \nu_q. \qquad \square$$

Example 3.11 (Ex. 3.2 continued) Figure 3 shows a tree together with a run κ. We compute the weight $\langle\kappa\rangle$ of this run as follows:

$$\begin{aligned}
\langle\kappa\rangle &= \langle\kappa|_1\rangle \circ \langle\kappa|_2\rangle \circ \delta(q_1q_2, \sigma, q_1) \\
&= \delta(\varepsilon, \alpha, q_2) \circ \delta(\varepsilon, \alpha, q_1) \circ \delta(q_1, \gamma, q_1) \circ \delta(q_2q_1, \sigma, q_1) \circ \delta(\varepsilon, \alpha, q_2) \circ \delta(q_1q_2, \sigma, q_1) \\
&= \{\varepsilon\} \circ \{\varepsilon\} \circ \{1\} \circ \{2\} \circ \{\varepsilon\} \circ \{1\} \\
&= \{121\}
\end{aligned}$$

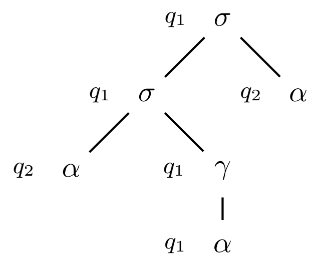

Figure 3: A tree together with a run.

Thus, the weight of κ is the position of the lowest node labeled α. Taking all runs on a tree ξ into account, it can be shown that $[\![\xi]\!]_{q_1} = \mathrm{pos}(\xi)$ and $[\![\xi]\!]_{q_2} = \{\varepsilon\}$. Hence, we obtain $\varphi_{\mathcal{A}} = \mathrm{pos}$. □

Definition 3.12 For every tree $\xi \in T_\Sigma(S^Q)$ and run $\kappa \in R(\xi)$ we call κ a *victorious run on* ξ if the following condition holds: $\langle\kappa\rangle = [\![\xi]\!]_{\kappa(\varepsilon)}$. □

The following observations are based on [BMV10]. Observation 3.13 states that there is a victorious run within a given set of runs if the semiring is extremal, whereas Observation 3.14 shows that the prefix of a victorious run is interchangeable with the prefix of any other victorious run.

Observation 3.13 *Let S be an extremal semiring. For every $\xi \in T_\Sigma(S^Q)$ and $q \in Q$ there is a $\kappa \in R^q(\xi)$ such that κ is victorious.* ■

Observation 3.14 *Let $\xi \in T_\Sigma(S^Q)$, $w \in \mathrm{pos}(\xi)$, and $\kappa \in R(\xi)$ victorious. Then we obtain $\langle\kappa\rangle = [\![(\langle\kappa|_w\rangle \cdot e_{\kappa(w)}) \cdot \xi[z]_w]\!]_{\kappa(\varepsilon)}$.*

Proof.

$$\begin{aligned}
&[\![(\langle\kappa|_w\rangle \cdot e_{\kappa(w)}) \cdot \xi[z]_w]\!]_{\kappa(\varepsilon)} \\
&= \textstyle\sum_{\kappa' \in R^{\kappa(\varepsilon)}\left((\langle\kappa|_w\rangle \cdot e_{\kappa(w)}) \cdot \xi[z]_w\right)} \langle\kappa'\rangle \\
&= \textstyle\sum_{\kappa' \in R^{\kappa(\varepsilon)}(\xi[z]_w), \kappa'(w) = \kappa(w)} \langle\langle\kappa|_w\rangle \cdot \kappa'\rangle \\
&= \textstyle\sum_{\kappa' \in R^{\kappa(\varepsilon)}(\xi[z]_w), \kappa'(w) = \kappa(w)} \langle\kappa|_w \cdot \kappa'\rangle \\
&= \langle\kappa\rangle\ .
\end{aligned}$$

For the last equation, we note that the summands on the left-hand side form a subset of $\{\langle\nu\rangle \mid \nu \in R^{\kappa(\varepsilon)}(\xi)\}$, which contains $\langle\kappa\rangle$. Since S is extremal and $\langle\kappa\rangle = [\![\xi]\!]_{\kappa(\varepsilon)}$, the equation holds. ■

3.2 The Twins Property

Definition 3.15 We define two binary relations over Q in the following way. For every two states $p, q \in Q$ we have

- $(p, q) \in \textsc{Siblings}(\mathcal{A})$ iff there is a tree $\xi \in T_\Sigma$ such that $[\![\xi]\!]_p \neq 0$ and $[\![\xi]\!]_q \neq 0$, and
- $(p, q) \in \textsc{Twins}(\mathcal{A})$ iff for every context $\zeta \in C_\Sigma$ we have that $[\![e_p \cdot \zeta]\!]_p \neq 0$ and $[\![e_q \cdot \zeta]\!]_q \neq 0$ implies $[\![e_p \cdot \zeta]\!]_p = [\![e_q \cdot \zeta]\!]_q$. □

Hence, states are siblings if there is at least one tree that can be decorated with either one of those states at its root and gets assigned non-zero weights in both cases. Moreover, we call states twins if the weights on contexts that are decorated with those states both at node z and at their root coincide for all contexts. The combination of both concepts results in the following definition of the twins property:

Definition 3.16 The wta $\mathcal{A}$ has the *twins property* if $\textsc{Siblings}(\mathcal{A}) \subseteq \textsc{Twins}(\mathcal{A})$. □

Example 3.17 We give three small examples for each of the above terms.

- Any acyclic wta has the twins property: for every $p \in Q$ and $\zeta \in C_\Sigma$ the weight of the context where both leaf z and its root are decorated with p is $[\![e_p \cdot \zeta]\!]_p = 0$.
- The wta from Example 3.2 has only two states. These states are siblings as they can be reached by processing the tree $\xi = \alpha$. The assigned weights are $\{\varepsilon\}$ in both cases. However, they are not twins because there exists at least one context that gets assigned different weights, for example $\zeta = \gamma(z)$: we have $[\![e_{q_1} \cdot \zeta]\!]_{q_1} = \{1\}$ but $[\![e_{q_2} \cdot \zeta]\!]_{q_2} = \{\varepsilon\}$.
- Consider the wta from Figure 4 over the semiring $\mathcal{S} = ([0, 1], +, \cdot, 0, 1)$. This automaton has the twins property: the pair $(q_1, q_2) \in \textsc{Siblings}(\mathcal{A})$ since $[\![\alpha]\!]_{q_1} \neq 0$ and $[\![\alpha]\!]_{q_2} \neq 0$. Both states are also twins as their transitions are symmetric. An equivalent deterministic wta is shown in Figure 5. □

Deciding the twins property requires us to be able to enumerate the set $\textsc{Siblings}(\mathcal{A})$ in finite time. The following observation shows that this is possible.

Observation 3.18 *If S is a zero-sum free semiring, the following statement holds:* $\textsc{Siblings}(\mathcal{A}) = \textsc{Sib}(\mathcal{A})$, *where* $\textsc{Sib}(\mathcal{A})$ *is defined like* $\textsc{Siblings}(\mathcal{A})$ *with the additional condition that* $\mathrm{ht}(\xi) < |Q|^2$.

Proof. The direction $\supseteq$ is trivial. Thus, we only show the direction $\subseteq$. This is done by contradiction. Let $p, q \in Q$ and $\xi \in T_\Sigma$ such that

(i) $(p, q) \in \textsc{Siblings}(\mathcal{A})$, i. e., $[\![\xi]\!]_p \neq 0$ and $[\![\xi]\!]_q \neq 0$,

(ii) $(p, q) \notin \textsc{Sib}(\mathcal{A})$, i. e., $\mathrm{ht}(\xi) \geq |Q|^2$.

We assume that ξ is the smallest counterexample and show that we find a smaller counterexample $\xi' \in T_\Sigma$, thus, contradicting that assumption. By (i) there are two runs $\kappa_p \in R^p(\xi)$ and $\kappa_q \in R^q(\xi)$ with weights $\langle \kappa_p \rangle \neq 0$ and $\langle \kappa_q \rangle \neq 0$. In addition, by $\mathrm{ht}(\xi) \geq |Q|^2$ we find two positions $w_1, w_2 \in \mathrm{pos}(\xi)$ such that w_1 is above w_2, $\kappa_p(w_1) = \kappa_p(w_2)$, and $\kappa_q(w_1) = \kappa_q(w_2)$.

At this point we cut out the slice between w_1 and w_2 by setting $\xi' = \xi[\xi|_{w_2}]_{w_1}$ and construct the runs κ'_p and κ'_q on ξ' as follows: $\kappa'_p = \kappa_p[\kappa_p|_{w_2}]_{w_1}$ and $\kappa'_q = \kappa_q[\kappa_q|_{w_2}]_{w_1}$.

Because κ_p and κ_q had non-zero weights we conclude $\langle\kappa'_p\rangle \neq 0$ and $\langle\kappa'_q\rangle \neq 0$. Moreover, since S ist zero-sum free we obtain $[\![\xi']\!]_p \neq 0$ and $[\![\xi']\!]_q \neq 0$. ■

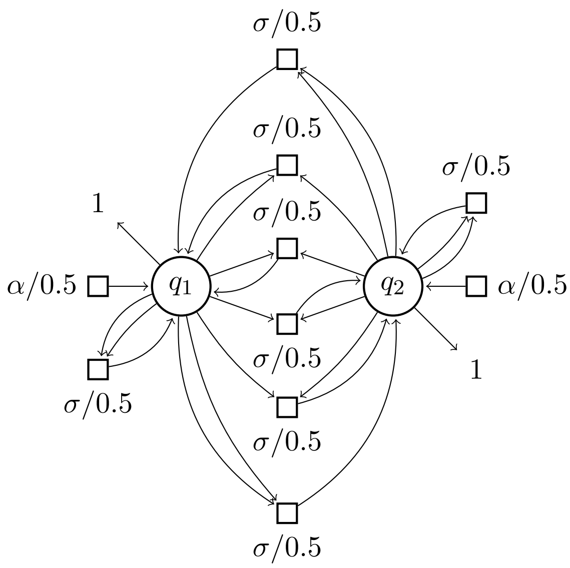

Figure 4: Siblings and twins

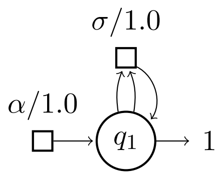

Figure 5: Equivalent deterministic wta

4 Decidability of the Twins Property

After having established some fundamentals in the previous two chapters, in this chapter we aim to prove the following theorem:

Theorem 4.1 *The twins property of wta over extremal semifields is decidable.*

Recall that extremal semifields always admit a maximal factorization whose existence is crucial to our later proofs. However, extremal semirings with certain characteristics can be transferred into extremal semifields; the twins property of a wta itself remains unaffected by this transformation. These characteristics include besides commutativity that the semiring is cancellative. Then, by simply adding the missing multiplicative inverse elements to the extremal semiring we obtain an extremal semifield. Henceforth, let S be an extremal semifield and (f, g) a maximal factorization.

Turning towards our theorem, it is quite easy to describe a naïve algorithm that searches for a context contradicting the twins property and terminates once such a counterexample has been found. However, if the given wta has the twins property there will be no counterexample. This results in a nonterminating algorithm, due to having an infinite number of contexts. Based on Kirsten's approach for weighted finite string automata [Kir12] we will compress our search space to a finite size if a given wta has the twins property. In the other case, the search space might remain infinite. This, however, does not turn out to be a problem, since we will be able to find a counterexample contradicting the twins property. Throughout the following sections we divide our proof into smaller parts, leading to a proof for Theorem 4.1 eventually.

This chapter is structured as follows: Section 4.1 rephrases the twins relation by an approach to run two wta in parallel. Section 4.2 deals with compressing the search space by applying a maximal factorization, and Section 4.3 suggests two decision algorithms, one proving Theorem 4.1 and the second one being a slight improvement of the first algorithm by avoiding redundant calculation.

4.1 Rephrasing the Twins Relation

The definition of $\textsc{Twins}(\mathcal{A})$ involves two vectors $[\![e_p \cdot \zeta]\!]$ and $[\![e_{\bar{q}} \cdot \zeta]\!]$ for every context $\zeta \in C_\Sigma$. Comparing both vectors requires a parallel execution of $\mathcal{A}$, which, unfortunately, is inconvenient for our later intent to extract a common factor from both vectors. Hence, we concatenate both vectors by constructing a *union wta* $\mathcal{A} \cup \bar{\mathcal{A}}$ that runs both instances of $\mathcal{A}$ in parallel (Figure 6).

To this end, let $\mathcal{A} = (Q, \Sigma, S, \delta, \nu)$ a wta and $\bar{\mathcal{A}} = (\bar{Q}, \Sigma, S, \bar{\delta}, \bar{\nu})$ the equivalent wta with $\bar{Q} = \{\bar{q} \mid q \in Q\}$, $\bar{\nu}_{\bar{q}} = \nu_q$ for every $q \in Q$, and $\bar{\delta}(\bar{q}_1 \ldots \bar{q}_k, \sigma, \bar{q}) = \delta(q_1 \ldots q_k, \sigma, q)$ for every $\sigma \in \Sigma^{(k)}$ and $q_1, \ldots, q_k, q \in Q$.

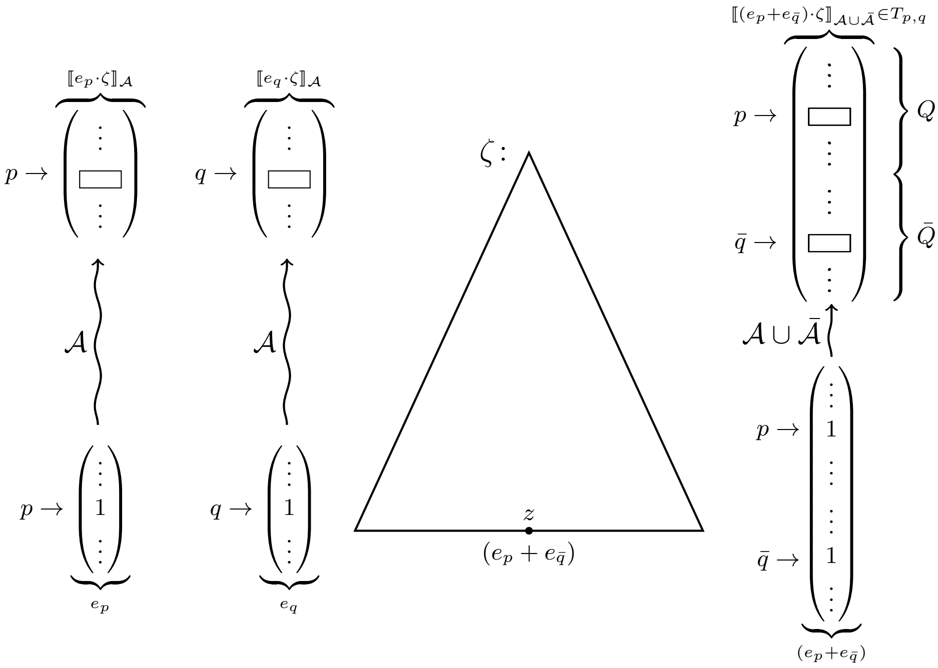

Figure 6: Moving from parallel execution of $\mathcal{A}$ (left-hand side) to the union wta $\mathcal{A}\cup\bar{\mathcal{A}}$ (right-hand side).

Definition 4.2 We construct $\mathcal{A}\cup\bar{\mathcal{A}} = (Q\cup\bar{Q}, \Sigma, S, \delta', \nu')$ with δ' and ν' as follows:

$$\delta'(q_1\dots q_k,\sigma,q) = \begin{cases} \delta(q_1\dots q_k,\sigma,q) & \text{if } q_1,\dots,q_k,q\in Q \\ \bar{\delta}(q_1\dots q_k,\sigma,q) & \text{if } q_1,\dots,q_k,q\in \bar{Q} \\ 0 & \text{otherwise} \end{cases}$$

and

$$\nu'_q = \begin{cases} \nu_q & \text{if } q\in Q \\ \bar{\nu}_q & \text{if } q\in\bar{Q} \end{cases}.$$

□

Based on this construction, we make the following two observation. Their proofs are shown in Appendix A.

Observation 4.3 *Let $u,v\in S^Q$ and $\zeta\in C_\Sigma$. Then we have $[\![(u+v)\cdot\zeta]\!] = [\![u\cdot\zeta]\!] + [\![v\cdot\zeta]\!]$.*

Observation 4.4 *Let p,q be two states of a wta $\mathcal{A}$. Then we have*

$$(p,q)\in \textsc{Twins}(\mathcal{A}) \Leftrightarrow (p,\bar{q})\in\textsc{Twins}(\mathcal{A}\cup\bar{\mathcal{A}}).$$

For the purpose of later anaysis we collect the weight vectors $[\![\xi]\!]_{\mathcal{A}\cup\bar{\mathcal{A}}}$ of all trees $\xi = (e_p+e_{\bar{q}})\cdot\zeta$, $\zeta\in C_\Sigma$, within a new set $T_{p,q}$. Note that both e_p and $e_{\bar{q}}$ are vectors of dimension $Q\cup\bar{Q}$ over S, i. e., $e_p, e_{\bar{q}}\in S^{Q\cup\bar{Q}}$.

Definition 4.5 For every $p, q \in Q$ we define $T_{p,q} \subseteq S^{Q \cup \bar{Q}}$ such that

$$T_{p,q} = \{[\![(e_p + e_{\bar{q}}) \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}} \mid \zeta \in C_\Sigma\}. \qquad \square$$

With the above definition we can rephrase the twins property using a simple observation on the characteristics of elements $u \in T_{p,q}$.

Observation 4.6 *Let $p, q \in Q$ two arbitrary states. Then $(p, q) \in \textsc{Twins}(\mathcal{A})$ iff for every $u \in T_{p,q}$ we have that*

$$u_p \neq 0 \textit{ and } u_{\bar{q}} \neq 0 \textit{ implies } u_p = u_{\bar{q}}.$$

Proof. "$\Rightarrow$": Let $(p, q) \in \textsc{Twins}(\mathcal{A})$ and $u \in T_{p,q}$ such that $u_p, u_{\bar{q}} \neq 0$. By definition there is a context $\zeta \in C_\Sigma$ such that $u = [\![(e_p + e_{\bar{q}}) \cdot \zeta]\!]$. Hence, we conclude

$$\begin{aligned}
u_p &= [\![(e_p + e_{\bar{q}}) \cdot \zeta]\!]_p \\
&= [\![e_p \cdot \zeta]\!]_p + [\![e_{\bar{q}} \cdot \zeta]\!]_p && \text{(Observation 4.3)} \\
&= [\![e_p \cdot \zeta]\!]_p && ([\![e_{\bar{q}} \cdot \zeta]\!]_p = 0, (*)) \\
&= [\![e_{\bar{q}} \cdot \zeta]\!]_{\bar{q}} && \text{(Observation 4.4)} \\
&= [\![e_p \cdot \zeta]\!]_{\bar{q}} + [\![e_{\bar{q}} \cdot \zeta]\!]_{\bar{q}} && ([\![e_p \cdot \zeta]\!]_{\bar{q}} = 0, (*)) \\
&= [\![(e_p + e_{\bar{q}}) \cdot \zeta]\!]_{\bar{q}} = u_{\bar{q}}. && \text{(Observation 4.3)}
\end{aligned}$$

"$\Leftarrow$": Let $p, q \in Q$ be states of $\mathcal{A}$ and let $\zeta \in C_\Sigma$ satisfying $[\![e_p \cdot \zeta]\!] \neq 0$ and $[\![e_{\bar{q}} \cdot \zeta]\!] \neq 0$. By definition $u = [\![(e_p + e_{\bar{q}}) \cdot \zeta]\!] \in T_{p,q}$. Thus, we have

$$\begin{aligned}
([\![e_p \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}})_p &= ([\![e_p \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}})_p + ([\![e_{\bar{q}} \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}})_p && (([\![e_{\bar{q}} \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}})_p = 0, (*)) \\
&= ([\![(e_p + e_{\bar{q}}) \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}})_p && \text{(Observation 4.3)} \\
&= u_p && \text{(by definition)} \\
&= u_{\bar{q}} && \text{(by assumption)} \\
&= ([\![(e_p + e_{\bar{q}}) \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}})_{\bar{q}} && \text{(by definition)} \\
&= ([\![e_p \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}})_{\bar{q}} + ([\![e_{\bar{q}} \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}})_{\bar{q}} && \text{(Observation 4.3)} \\
&= ([\![e_{\bar{q}} \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}})_{\bar{q}}. && (([\![e_p \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}})_{\bar{q}} = 0, (*))
\end{aligned}$$

Thus, $(p, \bar{q}) \in \textsc{Twins}(\mathcal{A} \cup \bar{\mathcal{A}})$, and by Observation 4.4 we obtain $(p, q) \in \textsc{Twins}(\mathcal{A})$.

Note that, whenever a transition over mixed states is involved in the computation of a tree's weight, we conclude that the tree's weight is zero based on the construction of δ' in Definition 4.2. Hence, the equations marked with $(*)$ hold. ∎

For every pair $(p, q) \in \textsc{Siblings}(\mathcal{A})$, a vector $u \in S^{Q \cup \bar{Q}}$ is called a *critical vector (for (p, q))* if it does not fulfill the centered implication of Observation 4.6. Any critical vector in $T_{p,q}$ thereby witnesses $(p, q) \notin \textsc{Twins}(\mathcal{A})$. Consequently, $\mathcal{A}$ has the twins property iff $T_{p,q}$ contains no critical vector for every $(p, q) \in \textsc{Siblings}(\mathcal{A})$. Deciding the twins property thus amounts to searching for a critical vector. The proof of the following observation is shown in Appendix A.

Observation 4.7 *A vector $u \in S^{Q \cup \bar{Q}}$, $u \neq \tilde{0}$, is a critical iff $f(u)$ is a critical vector.*

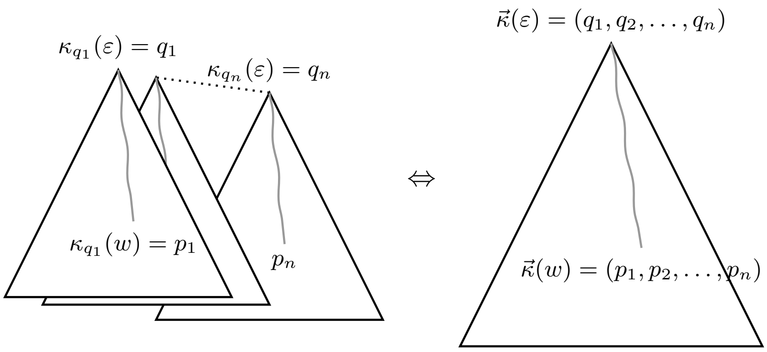

Figure 7: Combining simple runs $\kappa_{q_i} \in R^{q_i}((e_p + e_{\bar{q}}) \cdot \zeta)$ to a run vector $\vec{\kappa} \in R((e_p + e_{\bar{q}}) \cdot \zeta)^{Q \cup \bar{Q}}$.

4.2 Compressing the Search Space

In this section we approach the decidability of the twins property by compressing the previously defined set $T_{p,q}$, and, thus, compressing our search space for critical vectors. First, we modify Definition 5.13 and Lemma 5.14 from [BMV10, Section 5.4] to be also applicable to contexts rather than just trees.

Whenever we work with weight vectors $u \in S^{Q \cup \bar{Q}}$ of trees we assume that every entry has been produced by a certain victorious run. Hence, we combine all involved runs to a vector of runs (Figure 7).

Definition 4.8 For every $p, q \in Q$ and $\zeta \in C_\Sigma$ we define a set of vectors of runs on $\mathcal{A} \cup \bar{\mathcal{A}}$, denoted by $C_{p,q}(\zeta) \subseteq R((e_p + e_{\bar{q}}) \cdot \zeta)^{Q \cup \bar{Q}}$, in the following way: $\vec{\kappa} \in C_{p,q}(\zeta)$ iff

(i) $\vec{\kappa}_r \in R^r((e_p + e_{\bar{q}}) \cdot \zeta)$ for every $r \in Q \cup \bar{Q}$

(ii) for every pair $(w_1, w_2) \in \mathrm{pos}(\zeta) \times \mathrm{pos}(\zeta)$ with w_1 above w_2 and $\vec{\kappa}_r(w_1) = \vec{\kappa}_r(w_2)$ we have that $\vec{\kappa}_r|_{w_1}$ is victorious on $((e_p + e_{\bar{q}}) \cdot \zeta)|_{w_1}$. □

For evaluation purposes we define the following mapping on any run vector.

Definition 4.9 Let $p, q \in Q$ and $\zeta \in C_\Sigma$. For every $Q' \subseteq Q \cup \bar{Q}$ we define a mapping $\gamma_{Q'} \colon R((e_p + e_{\bar{q}}) \cdot \zeta)^{Q \cup \bar{Q}} \to S^{Q \cup \bar{Q}}$ that maps every vector of runs $\vec{\kappa}$ to a corresponding weight vector:

$$\forall q' \in Q \cup \bar{Q} \colon \gamma_{Q'}(\vec{\kappa})_{q'} = \begin{cases} \langle \vec{\kappa}_{q'} \rangle & \text{if } q' \in Q' \\ 0 & \text{otherwise.} \end{cases}$$ □

The next lemma proves the existence of a vector $\vec{\kappa} \in C_{p,q}(\zeta)$ as well as the equivalence of $\gamma_{Q'}(\vec{\kappa})$ for some $Q' \subseteq Q \cup \bar{Q}$ and the weight vector $[\![\xi]\!]$ of a tree $\xi = (e_p + e_{\bar{q}}) \cdot \zeta$. We will use this fact later on, when trying to compress our search space.

Lemma 4.10 *Let S be a commutative, extremal semiring and $\zeta \in C_\Sigma$. Then there is a $\vec{\kappa}$ such that $\vec{\kappa} \in C_{p,q}(\zeta)$ and there is a $Q' \subseteq Q \cup \bar{Q}$ such that $\gamma_{Q'}(\vec{\kappa}) = [\![(e_p + e_{\bar{q}}) \cdot \zeta]\!]$.*

Proof. As in [BMV10, Lemma 5.14] we prove the following statement by induction. For every $\zeta \in T_\Sigma \cup C_\Sigma$ and every $r \in Q \cup \bar{Q}$ there is a $\kappa \in R^r((e_p + e_{\bar{q}}) \cdot \zeta)$ such that $P(\zeta, \kappa)$, where

$$P(\zeta, \kappa) \Leftrightarrow \forall w \in \mathrm{pos}(\zeta)\colon \langle \kappa|_w \rangle = [\![((e_p + e_{\bar{q}}) \cdot \zeta)|_w]\!]_{\kappa|_w(\varepsilon)}.$$

Note that technically $(e_p + e_{\bar{q}}) \cdot \zeta$ is undefined for $\zeta \in T_\Sigma$. However, since we defined concatenation to be the replacement of node z within the second operand with the first operand we can simply extend our original definition in the following way:
For every $\xi, \zeta \in T_\Sigma \cup C_\Sigma$:

$$\xi \cdot \zeta = \begin{cases} \zeta[\xi]_w \text{ with } \zeta(w) = z & \text{if } \zeta \in C_\Sigma \\ \zeta & \text{if } \zeta \in T_\Sigma \end{cases}.$$

Induction base: We distinguish two cases.

(i) $\zeta = \alpha$, $\alpha \in \Sigma^{(0)}\colon \langle \kappa|_\varepsilon \rangle = \langle \kappa \rangle = [\![\alpha]\!]_{\kappa(\varepsilon)} = [\![((e_p + e_{\bar{q}}) \cdot \alpha)|_\varepsilon]\!]_{\kappa|_\varepsilon(\varepsilon)}$, and

(ii) $\zeta = z\colon \langle \kappa|_\varepsilon \rangle = \langle \kappa \rangle = [\![(e_p + e_{\bar{q}})]\!]_{\kappa(\varepsilon)} = [\![((e_p + e_{\bar{q}}) \cdot z)|_\varepsilon]\!]_{\kappa|_\varepsilon(\varepsilon)}$.

For the induction step we assume that $\zeta = \sigma(\xi_1, \ldots, \xi_{i-1}, \zeta_i, \xi_{i+1}, \ldots, \xi_k)$ with $\sigma \in \Sigma^{(k)}$, $\zeta_i \in T_\Sigma \cup C_\Sigma$ with $i \in \{1, \ldots, k\}$, and $\xi_\ell \in T_\Sigma$ for every $\ell \in \{1, \ldots, k\} \setminus \{i\}$.

By Observation 3.13 there is a $\kappa' \in R^r((e_p + e_{\bar{q}}) \cdot \zeta)$ such that κ' victorious. By our induction hypothesis there are victorious runs $\kappa_i \in R(((e_p + e_{\bar{q}}) \cdot \zeta)|_i)$ such that the predicate $P(((e_p + e_{\bar{q}}) \cdot \zeta)|_i, \kappa_i)$ holds for every $i \in \{1, \ldots, k\}$. Now we construct the run $\kappa \in R^r((e_p + e_{\bar{q}}) \cdot \zeta)$. For every $w \in \mathrm{pos}(\zeta)$ we have

$$\kappa(w) = \begin{cases} \kappa'(\varepsilon) & \text{if } w = \varepsilon \\ \kappa_i(w') & \text{if } w = iw', w' \in \mathrm{pos}(\zeta|_i), \text{ and } i \in \{1, \ldots, k\}. \end{cases}$$

Thus, $P(\zeta, \kappa)$ holds. Now we address $\gamma_{Q'}$: Let $Q' = \{q' \in Q \cup \bar{Q} \mid [\![(e_p + e_{\bar{q}}) \cdot \zeta]\!]_{q'} \neq 0\}$ and construct the run vector $\vec{\kappa} \in R((e_p + e_{\bar{q}}) \cdot \zeta)^{Q'}$ as follows.

$$\forall q \in Q'\colon \vec{\kappa}_q \in R^q((e_p + e_{\bar{q}}) \cdot \zeta) \text{ such that } P(\zeta, \vec{\kappa}_q).$$

Thus, we have

$$\gamma_{Q'}(\vec{\kappa})_q = \begin{cases} \langle \vec{\kappa}_q \rangle = [\![(e_p + e_{\bar{q}}) \cdot \zeta]\!]_q & \text{if } q \in Q' \\ 0 & \text{otherwise.} \end{cases}$$

and hence, $\gamma_{Q'}(\vec{\kappa}) = [\![(e_p + e_{\bar{q}}) \cdot \zeta]\!]$. ∎

Now that we have all the prerequisites needed to prove that all vectors in $T_{p,q}$ are scalar multiples of a finite set of vectors, we aim to do so in the following lemma.

Lemma 4.11 *Let S be a commutative, extremal semiring. Assume that $\mathcal{A}$ has the twins property. Then there is a finite set $S' \subseteq S^{Q \cup \bar{Q}}$ such that for every $(p, q) \in$ SIBLINGS$(\mathcal{A})$ we have*

$$T_{p,q} \subseteq S \cdot S'.$$

PROOF. We construct sets $S', S'' \subseteq S^{Q \cup \bar{Q}}$ and show the following inclusions:

$$T_{p,q} \subseteq S \cdot S'' \subseteq S \cdot S'. \qquad (*)$$

Let $S'' = \{\gamma_{Q'}(\vec{\kappa}) \mid (p, q) \in \text{SIBLINGS}(\mathcal{A}), \zeta \in C_\Sigma, \vec{\kappa} \in C_{p,q}(\zeta), Q' \subseteq Q \cup \bar{Q}\}$ and S' defined exactly as S'' with the additional condition $\text{ht}(\zeta) < 2|Q|^{2|Q|}$.

The first inclusion of $(*)$ is proved by Lemma 4.10. Hence, we only have to show the second inclusion. Let $s \in S$, $(p, q) \in$ SIBLINGS$(\mathcal{A})$, $\zeta \in C_\Sigma$, $\vec{\kappa} \in C_{p,q}(\zeta)$, and $Q' \subseteq Q \cup \bar{Q}$ such that

$$s \cdot \gamma_{Q'}(\vec{\kappa}) \notin S \cdot S'. \qquad (**)$$

Because of the definition of S' we conclude $\text{ht}(\zeta) \geq 2|Q|^{2|Q|}$.

Let $Q' = \{q' \in Q \cup \bar{Q} \mid [\![(e_p + e_{\bar{q}}) \cdot \zeta]\!]_{q'} \neq 0\}$ which yields $\langle \vec{\kappa}_r \rangle \neq 0$ for every $r \in Q \cup \bar{Q}$, and assume that ζ is the smallest context that fulfills $(**)$. We construct a smaller context ζ' and a run vector $\kappa' \in C_{p,q}(\zeta')$ such that ζ' is obtained from ζ by cutting out a slice and $s \cdot \gamma_{Q'}(\vec{\kappa}) = s \cdot s' \cdot \gamma_{Q'}(\vec{\kappa}')$ with $s' \in S$. Then, if $s \cdot s' \cdot \gamma_{Q'}(\vec{\kappa}') \in S \cdot S'$, so is the left-hand side of the equation, i. e., ζ did not fulfill $(**)$. On the other hand, ζ was not the smallest counterexample, contradicting our assumption.

To this end, let $w \in \text{pos}(\zeta)$ be the position of node z. We are able two find a pair of positions $(w_1, w_2) \in \text{pos}(\zeta) \times \text{pos}(\zeta)$ such that $\vec{\kappa}(w_1) = \vec{\kappa}(w_2)$ and both w_1 and w_2 are either above or below w. This leads to the following two cases as depicted in Figure 8.

(a) $|w| \leq |Q|^{2|Q|}$: Any path of a length of at least $2|Q|^{2|Q|}$ shares a common prefix of length at most $|Q|^{2|Q|}$ with the path leading to w. Thus, there remain at least $|Q|^{2|Q|}+1$ positions on that path, and by the pidgeonhole principle we find (w_1, w_2).

(b) $|w| > |Q|^{2|Q|}$: The path leading to w contains at least $|Q|^{2|Q|} + 1$ positions and again, by the pidgeonhole principle we find the pair (w_1, w_2) on that path.

Now, we pick a pair (w_1, w_2) of positions such that the length of w_1 is minimal and cut out the slice between w_1 and w_2 yielding the smaller context ζ' $\zeta[\zeta|_{w_2}]_{w_1}$. In addition, we construct $\vec{\kappa}'$ such that $\vec{\kappa}'_r = \vec{\kappa}_r[\vec{\kappa}_r|_{w_2}]_{w_1}$ for every $r \in Q \cup \bar{Q}$. Because of our choice of w_1 we have that $\vec{\kappa}' \in C_{p,q}(\zeta')$. Finally, we use the twins property to show that there is an $s' \in S$ such that $\gamma_{Q'}(\vec{\kappa}) = s' \cdot \gamma_{Q'}(\vec{\kappa}')$. If $Q' = \emptyset$, we choose $s' = 0$. Otherwise we choose an arbitrary state $r' \in Q'$ and set $s' = [\![e_{\vec{\kappa}_{r'}(w_2)} \cdot \zeta'']\!]$ where $\zeta'' = \zeta[z]_{w_2}|_{w_1}$ is the slice that has been cut out. Let $r \in Q'$, $p' = \vec{\kappa}_r(w_1) = \vec{\kappa}_r(w_2)$ and $q' = \vec{\kappa}_{r'}(w_1) = \vec{\kappa}_{r'}(w_2)$.

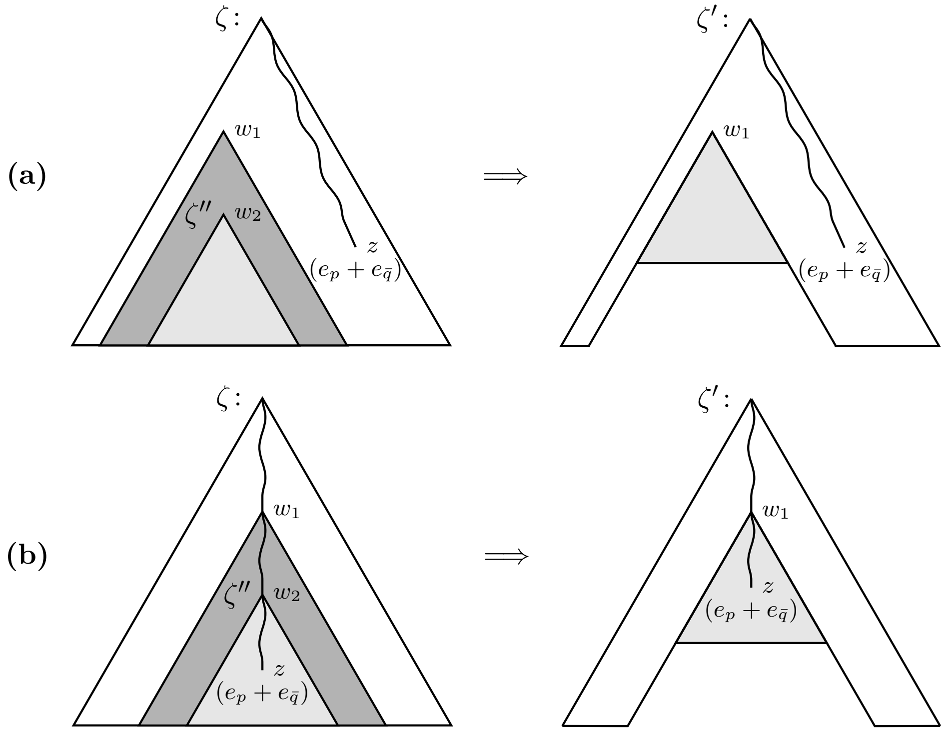

Figure 8: Two cases for the construction of $\zeta' = \zeta[\zeta|_{w_2}]_{w_1}$.

Then, we have

$$
\begin{aligned}
\gamma_{Q'}(\vec{\kappa})_r &= \langle \vec{\kappa}_r \rangle = \langle \langle \vec{\kappa}_r|_{w_1} \rangle \cdot \vec{\kappa}_r[z]_{w_1} \rangle \\
&= \langle [\![(\langle \vec{\kappa}_r|_{w_2} \rangle \cdot e_{p'}) \cdot \zeta'']\!]_{p'} \cdot \vec{\kappa}_r[z]_{w_1} \rangle && \text{(Observation 3.14)} \\
&= \langle \vec{\kappa}_r|_{w_2} \rangle \cdot [\![e_{p'} \cdot \zeta'']\!]_{p'} \cdot \langle 1 \cdot \vec{\kappa}_r[z]_{w_1} \rangle && \text{(commutativity)} \\
&= \langle \vec{\kappa}_r|_{w_2} \rangle \cdot [\![e_{q'} \cdot \zeta'']\!]_{q'} \cdot \langle 1 \cdot \vec{\kappa}_r[z]_{w_1} \rangle && (\dagger) \\
&= s' \cdot \langle \langle \vec{\kappa}_r|_{w_2} \rangle \cdot \vec{\kappa}_r[z]_{w_1} \rangle && \text{(commutativity)} \\
&= s' \cdot \langle \vec{\kappa}'_r \rangle && \text{(Observation 3.8)} \\
&= s' \cdot \gamma_{Q'}(\vec{\kappa}')_r \ .
\end{aligned}
$$

At ($\dagger$) we have used the twins property. This is only possible if $(p', q') \in \textsc{Siblings}(\mathcal{A} \cup \bar{\mathcal{A}})$. Thus, we prove the siblings relation of p' and q' by distinguishing two cases.

(i) z occurs in $\zeta[z]_{w_1}$: By our choice of Q' we have $\langle \vec{\kappa}_r \rangle \neq 0$ and $\langle \vec{\kappa}_{r'} \rangle \neq 0$ implying $\langle \vec{\kappa}_r|_{w_2} \rangle \neq 0$ and $\langle \vec{\kappa}_{r'}|_{w_2} \rangle \neq 0$. Thus, we get $[\![\zeta|_{w_2}]\!]_{p'} \neq 0$ and $[\![\zeta|_{w_2}]\!]_{q'} \neq 0$.

(ii) z occurs in $\zeta|_{w_2}$: Due to $(p, q) \in \textsc{Siblings}(\mathcal{A} \cup \bar{\mathcal{A}})$ there is a tree ξ such that $[\![\xi]\!]_p \neq 0$ and $[\![\xi]\!]_q \neq 0$. Again, by our choice of Q' we derive $\langle \vec{\kappa}_r|_{w_2} \rangle \neq 0$ and $\langle \vec{\kappa}_{r'}|_{w_2} \rangle \neq 0$. Since S is extremal, and therefore, zero-sum free, we obtain $[\![\xi \cdot \zeta]\!]_{p'} \neq 0$ and $[\![\xi \cdot \zeta]\!]_{q'} \neq 0$.

Algorithm 1 Decision algorithm

Require: $\mathcal{A} = (Q, \Sigma, S, \delta, \nu)$ a wta, S commutative, extremal, (f, g) max. factorization
Ensure: print "yes" iff $\mathcal{A}$ has the twins property

```
compute SIBLINGS(A)
for (p, q) ∈ SIBLINGS(A) in parallel do
    for u ∈ f(T_{p,q} \ {0̃}) do
        if u is a critical vector then
            print "no" and terminate
print "yes"
```

By the twins property of $\mathcal{A} \cup \bar{\mathcal{A}}$ we have $(p', q') \in \textsc{Twins}(\mathcal{A} \cup \bar{\mathcal{A}})$. Thus, using $\langle \vec{\kappa}_r \rangle \neq 0$, $\langle \vec{\kappa}_{r'} \rangle \neq 0$ again yields $\langle \vec{\kappa}_r[z]_{w_2}|_{w_1} \rangle \neq 0$ and $\langle \vec{\kappa}_{r'}[z]_{w_2}|_{w_1} \rangle \neq 0$; which leads to $[\![e_{p'} \cdot \zeta'']\!]_{p'} \neq 0$, $[\![e_{p'} \cdot \zeta'']\!]_{p'} \neq 0$. ∎

Here we use our factorization (f, g) over S and Observation 4.7. Thus, we obtain a compressed search space for critical vectors by applying the factorization to every set $T_{p,q}$ with $(p, q) \in \textsc{Siblings}(\mathcal{A})$.

Lemma 4.12 *Let (f, g) be a maximal factorization of dimension $Q \cup \bar{Q}$. Assume that $\mathcal{A}$ has the twins property. For every $(p, q) \in \textsc{Siblings}(\mathcal{A})$ the set $f(T_{p,q} \setminus \{\tilde{0}\})$ is finite.*

Proof. By Lemma 4.11 there is a finite set S' with

$$f(T_{p,q} \setminus \{\tilde{0}\}) \subseteq f(S \cdot S') \subseteq f(S') ,$$

where we used that (f, g) is maximal. Since S' is finite, so is $f(T_{p,q} \setminus \{\tilde{0}\})$. ∎

4.3 Two Decision Algorithms

Using Lemma 4.12 we can now state a simple decision algorithm that decides the twins property of a given wta.

Proof (of Theorem 4.1). At first, Algorithm 1 computes the set $\textsc{Siblings}(\mathcal{A})$. By Observation 3.18 this is possible. Then, for every pair (p, q) of siblings the set $f(T_{p,q} \setminus \{\tilde{0}\})$ is enumerated and checked for the occurrence of critical vectors. This process is done in parallel for all pairs of siblings, otherwise the algorithm would not terminate if there was a pair $(p, q) \in \textsc{Siblings}(\mathcal{A})$ such that $f(T_{p,q} \setminus \{\tilde{0}\})$ is infinite but does not contain a critical vector. At this point, there are two possible outcomes:

(a) the wta $\mathcal{A}$ has the twins property. In this case the sets $f(T_{p,q} \setminus \{\tilde{0}\})$ are finite. The algorithm will not find any critical vectors, and, thus, terminates once all sets are fully enumerated. The output in this case is "yes".

(b) the wta $\mathcal{A}$ does not have the twins property. Then, by Observation 4.6 the algorithm will find a critical vector at some point and terminate by outputting "no". ∎

Algorithm 2 Improved decision algorithm

Require: $\mathcal{A} = (Q, \Sigma, S, \delta, \nu)$ a wta, S commutative, extremal, (f, g) max. factorization
Ensure: print "yes" iff $\mathcal{A}$ has the twins property

```
compute SIBLINGS(A)
(T, C) ← (∅, ∅)
repeat
    (T′, C′) ← (T, C)
    (T, C) ← F(T′, C′)                              ▷ uses SIBLINGS(A)
until C contains critical vector or C = C′
if critical vector has been found then
    print "no"
else
    print "yes"
```

However, Algorithm 1 calculates certain weights repeatedly. It basically enumerates the infinite set C_Σ and computes $f[\![(e_p + e_{\bar{q}}) \cdot \zeta]\!]$ for each $\zeta \in C_\Sigma$ which results in the enumeration of the set $\bigcup_{(p,q) \in \textsc{Siblings}(\mathcal{A})} f(T_{p,q} \setminus \{\tilde{0}\})$. This approach computes the weight of each tree $(e_p + e_{\bar{q}}) \cdot \zeta$ from scratch, discarding previously obtained weights of subtrees and subcontexts.

Therefore, we show an improved algorithm. Algorithm 2 does not enumerate C_Σ explicitly. It rather pursues the enumeration of weight vectors. Once computed, the weight vectors can be reused, which prevents redundant calculation. The algorithm maintains a pair of subsets of $S^{Q \cup \bar{Q}}$. It starts with the pair $(\emptyset, \emptyset)$ and adds vectors by applying a monotone operation F until a critical vector is found within the pair's second component or no new vectors are added to that second component.

Definition 4.13 We define the unary operation $F \colon \mathcal{P}(S^{Q \cup \bar{Q}}) \times \mathcal{P}(S^{Q \cup \bar{Q}}) \to \mathcal{P}(S^{Q \cup \bar{Q}}) \times \mathcal{P}(S^{Q \cup \bar{Q}})$ such that $F((T, C)) = (T', C')$, and T', C' contain exactly the following elements:

(F1) for every $k \geq 0$, $\sigma \in \Sigma^{(k)}$, and $u_1, \ldots, u_k \in T$, if $[\![\sigma(u_1, \ldots, u_k)]\!] \neq \tilde{0}$, then $f[\![\sigma(u_1, \ldots, u_k)]\!] \in T'$,

(F2) for every $(p, q) \in \textsc{Siblings}(\mathcal{A})$, we have $f(e_p + e_{\bar{q}}) \in C'$,

(F3) for every $k \geq 1$, $\sigma \in \Sigma^{(k)}$, $i \in \{1, \ldots, k\}$, $u_i \in C$, and $u_\ell \in T$ for every $\ell \in \{1, \ldots, k\} \setminus \{i\}$, if $[\![\sigma(u_1, \ldots, u_k)]\!] \neq \tilde{0}$, then $f[\![\sigma(u_1, \ldots, u_k)]\!] \in C'$. □

We abbreviate the set $\mathcal{P}(S^{Q \cup \bar{Q}}) \times \mathcal{P}(S^{Q \cup \bar{Q}})$ with M, and define the partial order $\sqsubseteq$, i. e., a reflexive, transitive, and antisymmetric relation on M, such that for every two elements $(T, C), (T', C') \in M$ the following equivalence holds:

$$(T, C) \sqsubseteq (T', C') \quad \Leftrightarrow \quad T \subseteq T' \wedge C \subseteq C'.$$

Hence, M with $\sqsubseteq$ is a partially ordered set (poset). Note that the set $B := \mathcal{P}(S^{Q \cup \bar{Q}})$ with set inclusion is also a poset. Let $X \subseteq B$ a countable chain in B, i. e., a countable subset of B that only contains pairwise comparable elements. Since B is a power set with respect to set inclusion it is ω-complete, i. e., every countable chain has a supremum. The supremum of any countable chain X can be specified by sup $X = \bigcup_{x \in X} x$. Using the ω-completeness of B we conclude that M is also ω-complete: let $Y \subseteq M$ an arbitrary countable chain in M. Thus, $\sup Y = (\sup \{T_i \mid i \geq 0\}, \sup \{C_i \mid i \geq 0\}) = (\bigcup_{i \geq 0} T_i, \bigcup_{i \geq 0} C_i)$, where $(T_i, C_i) \in Y$ for every $i \geq 0$.

In addition to the ω-completeness of M we make the following two observations. The proofs thereof are shown in Appendix A.

Observation 4.14 *F is monotone on every countable chain $X \subseteq M$.*

Observation 4.15 *F is ω-continuous, i. e., for every nonempty countable chain in M that has a supremum, the supremum of $F(X)$ exists and $F(\sup X) = \sup F(X)$.*

The ω-completeness of M and Observations 4.14 and 4.15 are the requirements for Kleene's fixpoint theorem [Wec92, Section 1.5.2, Theorem 7]. It states that there is a least fixpoint (lfp) for the ω-continuous operator F that maps the ω-complete set M into itself, and it can be calculated as listed above. To prove both correctness and termination of our proposed algorithm we need to show that this fixpoint of F contains the set $\bigcup_{(p,q) \in \textsc{Siblings}(\mathcal{A})} f(T_{p,q} \setminus \{\tilde{0}\})$ (Lemma 4.17). That proof requires us to often use the following two statements.

Observation 4.16 *Let S be commutative and (f, g) maximal. Then for every $k \geq 0$, $\sigma \in \Sigma^{(k)}$, and $\xi_1, \ldots, \xi_k \in T_\Sigma(S^Q)$, we have that*

(i) $[\![\sigma(\xi_1, \ldots, \xi_k)]\!] = [\![\sigma([\![\xi_1]\!], \ldots, [\![\xi_k]\!])]\!]$ *and*

(ii) $f[\![\sigma([\![\xi_1]\!], \ldots, [\![\xi_k]\!])]\!] = f[\![\sigma(f[\![\xi_1]\!], \ldots, f[\![\xi_k]\!])]\!]$.

Proof. By [FV09, Section 3.2] and [BMV10, Lemma 5.5], respectively. ■

Lemma 4.17 *Let $T^f, C^f \in \mathcal{P}(S^{Q \cup \bar{Q}})$ such that*

(i) $T^f = f([\![T_\Sigma]\!] \setminus \{\tilde{0}\})$ *and*

(ii) $C^f = \bigcup_{(p,q) \in \textsc{Siblings}(\mathcal{A})} f(T_{p,q} \setminus \{\tilde{0}\})$.

Then (T, C) is the least fixpoint of F, i. e., $(T^f, C^f) = \text{lfp } F$.

Proof. We have to prove the following two directions.

(a) "$(T^f, C^f) \sqsubseteq \text{lfp } F$": Let $(\hat{T}, \hat{C}) = F(\hat{T}, \hat{C})$, i. e., $(\hat{T}, \hat{C})$ is an arbitrary fixpoint of F. By proving $(T^f, C^f) \sqsubseteq F(\hat{T}, \hat{C})$ we show that (T^f, C^f) is smaller (in the sense of the partial order $\sqsubseteq$) than every fixpoint of F, and, hence, $(T^f, C^f) \sqsubseteq \text{lfp } F$.

We show $(T^f, C^f) \sqsubseteq F(\hat{T}, \hat{C})$ by contradiction.

Let $\xi \in T_\Sigma$ the smallest tree such that $[\![\xi]\!] \neq \tilde{0}$, and $u = f[\![\xi]\!]$ such that $u \in T^f$ but $u \notin \hat{T}$. By definition of T_Σ there are $k \geq 0, \sigma \in \Sigma^{(k)}$, and $\xi_1, \dots, \xi_k \in T_\Sigma$ such that $\xi = \sigma(\xi_1, \dots, \xi_k)$. Hence, we derive

$$\begin{aligned} u &= f[\![\xi]\!] \\ &= f[\![\sigma(\xi_1, \dots, \xi_k)]\!] \\ &= f[\![\sigma([\![\xi_1]\!], \dots, [\![\xi_k]\!])]\!] \\ &= f[\![\sigma(\underbrace{f[\![\xi_1]\!]}_{u_1}, \dots, \underbrace{f[\![\xi_k]\!]}_{u_k})]\!]. \end{aligned}$$

By $[\![\xi]\!] \neq \tilde{0}$ we know that $u_i \neq \tilde{0}$ for every $i \in \{1, \dots, k\}$. Thus, by definition $u_i \in T^f$ holds for every u_i. Now we have to distinguish two cases. Either every u_i is also an element of $\hat{T}$, but then so is u once we apply $(F1)$ to $(\hat{T}, \hat{C})$ and use that $(\hat{T}, \hat{C})$ is a fixpoint, or there is a u_j, $j \in \{1, \dots, k\}$ such that $u_j \notin \hat{T}$. In the latter case ξ was not the smallest counterexample which contradicts our assumption. Thus, we conclude $T^f \subseteq \hat{T}$.

Let $(p, q) \in \textsc{Siblings}(\mathcal{A})$ and $\zeta \in C_\Sigma$ the smallest context such that $[\![(e_p + e_{\bar{q}}) \cdot \zeta]\!] \neq \tilde{0}$, and $u = f[\![(e_p + e_{\bar{q}}) \cdot \zeta]\!]$ such that $u \in C^f$ but $u \notin \hat{C}$. By definition of C_Σ there are $k \geq 0, \sigma \in \Sigma^{(k)}$, $\zeta_i \in C_\Sigma$, $i \in \{1, \dots, k\}$, and $\xi_\ell \in T_\Sigma$ for every $\ell \in \{1, \dots, k\} \setminus \{i\}$ such that $\zeta = \sigma(\xi_1, \dots, \zeta_i, \dots, \xi_k)$. Hence, we derive

$$\begin{aligned} u &= f[\![(e_p + e_{\bar{q}}) \cdot \zeta]\!] \\ &= f[\![(e_p + e_{\bar{q}}) \cdot \sigma(\xi_1, \dots, \zeta_i, \dots, \xi_k)]\!] \\ &= f[\![\sigma(\xi_1, \dots, (e_p + e_{\bar{q}}) \cdot \zeta_i, \dots, \xi_k)]\!] \\ &= f[\![\sigma([\![\xi_1]\!], \dots, [\![(e_p + e_{\bar{q}}) \cdot \zeta_i]\!], \dots, [\![\xi_k]\!])]\!] \\ &= f[\![\sigma(\underbrace{f[\![\xi_1]\!]}_{u_1}, \dots, \underbrace{f[\![(e_p + e_{\bar{q}}) \cdot \zeta_i]\!]}_{u_i}, \dots, \underbrace{f[\![\xi_k]\!]}_{u_k})]\!]. \end{aligned}$$

By $[\![(e_p + e_{\bar{q}}) \cdot \zeta]\!] \neq \tilde{0}$ we know that $u_\ell \neq \tilde{0}$ for every $\ell \in \{1, \dots, k\} \setminus \{i\}$ and $u_i \neq \tilde{0}$. Thus, for every u_ℓ we have that $u_\ell \in T^f$ and $u_i \in C^f$ by the definitions of T^f and C^f. We already proved $T^f \subseteq \hat{T}$. Hence every u_ℓ is an element of $\hat{T}$. Again, we need to distinguish two cases: $u_i \in \hat{C}$, but then so is u because of $(F3)$, or $u_i \notin \hat{C}$. In this case ζ was not the smallest counterexample. This contradicts our assumption, and we conclude $C^f \subseteq \hat{C}$.

(b) "lfp $F \sqsubseteq (T^f, C^f)$": We show that (T^f, C^f) is a prefixpoint, i. e., an element $m \in M$ such that $F(m) \sqsubseteq m$. By Park's Theorem [Wec92, Section 1.5.2, Proposition 9] we obtain that lfp $F \sqsubseteq m$.

We define $(\hat{T}, \hat{C}) := F(T^f, C^f)$. Let $u \in \hat{T}$. It is a result of the application of $(F1)$ to (T^f, C^f). Hence, there are $k \geq 0$, $\sigma \in \Sigma^{(k)}$, $u_1, \dots, u_k \in T^f$ such that $[\![\sigma(u_1, \dots, u_k)]\!] \neq \tilde{0}$, and we have $u = f[\![\sigma(u_1, \dots, u_k)]\!]$. We obtain $u_i = f[\![\xi_i]\!] \neq \tilde{0}$ for every $i \in \{1, \dots, k\}$ and $\xi_i \in T_\Sigma$ because every u_i is an element of T^f and

$[\![\sigma(u_1, \ldots, u_k)]\!] \neq \tilde{0}$. Hence, we derive

$$\begin{aligned} u &= f[\![\sigma(u_1, \ldots, u_k)]\!] \\ &= f[\![\sigma(f[\![\xi_1]\!], \ldots, f[\![\xi_k]\!])]\!] \\ &= f[\![\sigma([\![\xi_1]\!], \ldots, [\![\xi_k]\!])]\!] \\ &= f[\![\sigma(\xi_1, \ldots, \xi_k)]\!], \end{aligned}$$

which is an element of T^f by definition. Hence, $\hat{T} \subseteq T^f$.

Now let $u \in \hat{C}$. Then there are two different possibilities.

(b1) As a result of $(F2)$ we have that $u = f(e_p + e_{\bar{q}})$ for some $(p, q) \in \textsc{Siblings}(\mathcal{A})$. By definition of $T_{p,q}$ we have $[\![e_p + e_{\bar{q}}]\!] \in T_{p,q}$ if $\zeta = z$. Hence, we conclude that $f[\![e_p + e_{\bar{q}}]\!] = f(e_p + e_{\bar{q}}) = u$ is an element of C^f.

(b2) Otherwise $u = f[\![\sigma(u_1, \ldots, u_k)]\!]$ is a result of applying $(F3)$ to (T^f, C^f). and there are $k \geq 0$, $\sigma \in \Sigma^{(k)}$, $u_i \in C^f$, $i \in \{1, \ldots, k\}$, $u_\ell \in T^f$ for every $\ell \in \{1, \ldots, k\} \setminus \{i\}$ such that $[\![\sigma(u_1, \ldots, u_k)]\!] \neq \tilde{0}$. Using $[\![\sigma(u_1, \ldots, u_k)]\!] \neq \tilde{0}$ and the definitions of T^f and C^f we have that every $u_\ell = f[\![\xi_\ell]\!] \neq \tilde{0}$ and $u_i = f[\![(e_p + e_{\bar{q}}) \cdot \zeta]\!] \neq \tilde{0}$, where $\zeta \in C_\Sigma$. Hence, we conclude $u \in C^f$ by deriving

$$\begin{aligned} u &= f[\![\sigma(u_1, \ldots, u_i, \ldots, u_k)]\!] \\ &= f[\![\sigma(f[\![\xi_1]\!], \ldots, f[\![(e_p + e_{\bar{q}}) \cdot \zeta]\!], \ldots, f[\![\xi_k]\!])]\!] \\ &= f[\![\sigma([\![\xi_1]\!], \ldots, [\![(e_p + e_{\bar{q}}) \cdot \zeta]\!], \ldots, [\![\xi_k]\!])]\!] \\ &= f[\![\sigma(\xi_1, \ldots, (e_p + e_{\bar{q}}) \cdot \zeta, \ldots, \xi_k)]\!] \\ &= f\,\underbrace{[\![(e_p + e_{\bar{q}}) \cdot \underbrace{\sigma(\xi_1, \ldots, \zeta, \ldots, \xi_k)}_{\in C_\Sigma}]\!]}_{\in T_{p,q} \setminus \{\tilde{0}\}}\,. \end{aligned}$$

Both cases prove that $\hat{C} \subseteq C^f$. ∎

5 Conclusion

In this thesis we proved the decidability for weighted tree automata over extremal semifields. The basic idea of this proof, that is to say compressing the search space for critical vectors, is adopted from Kirsten's decidability proof for weighted string automata [Kir12].

In doing so, we had to generalize Kirsten's approach in two different ways: (a) We permit the use of arbitrary extremal semifields by making use of maximal factorizations. The application thereof is also present in Kirsten's work; it is, however, only used in an implicit way. (b) We consider weighted tree automata instead of weighted string automata. Because of the arising need to distinguish between trees and contexts the proof turns out to be more complex.

The question as to which of our decision algorithms performs better remains open, as the complexity of neither algorithm has been investigated. However, transferring Kirsten's result that deciding the twins property is PSPACE-hard, presents a lower bound on our algorithms' complexity.

Moreover, the height restriction of the set S' within Lemma 4.11 can be lowered to $\mathrm{ht}(\zeta) < 2|Q|^2$, as Algorithm 1 only needs the p- and $\bar{q}$-components of vectors $u \in T_{p,q}$ to search for critical vectors. This modification might possibly improve the algorithm's performance.

5.1 Future Work

Future research should fathom possibilities and sufficient requirements to determinize weighted tree automata over arbitrary, not necessarily extremal, semifields. In practical applications of machine translation weighted tree automata over extremal semirings or semifields are rarely used. The use of the semiring of non-negative reals for example, is far more likely. The determinization of wta over arbitrary semirings would not just approximate a sentence's or translation's likelihood with its best derivation; instead, it would cumulate all its derivations.

Appendix A

Additional Proofs

Proof of Observation 3.8

Proof. By induction on ζ. Let ξ, ζ, κ, and κ' as in the observation. For the induction base let $\zeta = z$.

$$\langle \kappa \cdot \kappa' \rangle = \langle \kappa \rangle = \langle \langle \kappa \rangle \rangle = \langle \langle \kappa \rangle \cdot \kappa' \rangle$$

For the induction step we have to distinguish the following two cases:

(i) $\zeta = \sigma(\xi_1, \dots, \xi_{i-1}, z, \xi_{i+1}, \dots, \xi_k)$ where $\sigma \in \Sigma^{(k)}$ and $\xi_1, \dots, \xi_k \in T_\Sigma$:

$$\begin{aligned}
\langle \kappa \cdot \kappa' \rangle &= \langle (\kappa \cdot \kappa')|_1 \rangle \cdot \dots \cdot \langle (\kappa \cdot \kappa')|_i \rangle \cdot \dots \cdot \langle (\kappa \cdot \kappa')|_k \rangle \\
&\quad \cdot \delta((\kappa \cdot \kappa')(1) \dots (\kappa \cdot \kappa')(i) \dots (\kappa \cdot \kappa')(k), \sigma, (\kappa \cdot \kappa')(\varepsilon)) \\
&= \langle \kappa'|_1 \rangle \cdot \dots \cdot \langle \kappa \rangle \cdot \dots \cdot \langle \kappa'|_k \rangle \cdot \delta(\kappa'(1) \dots q \dots \kappa'(k), \sigma, q') \\
&= \langle \kappa'|_1 \rangle \cdot \dots \cdot \langle \langle \kappa \rangle \rangle \cdot \dots \cdot \langle \kappa'|_k \rangle \cdot \delta(\kappa'(1) \dots q \dots \kappa'(k), \sigma, q') \\
&= \langle (\langle \kappa \rangle \cdot \kappa')|_1 \rangle \cdot \dots \cdot \langle (\langle \kappa \rangle \cdot \kappa')|_i \rangle \cdot \dots \cdot \langle (\langle \kappa \rangle \cdot \kappa')|_k \rangle \\
&\quad \cdot \delta((\langle \kappa \rangle \cdot \kappa')(1) \dots (\langle \kappa \rangle \cdot \kappa')(i) \dots (\langle \kappa \rangle \cdot \kappa')(k), \sigma, (\langle \kappa \rangle \cdot \kappa')(\varepsilon)) \\
&= \langle \langle \kappa \rangle \cdot \kappa' \rangle
\end{aligned}$$

(ii) $\zeta = \sigma(\xi_1, \dots, \xi_{i-1}, \zeta_i, \xi_{i+1}, \dots, \xi_k)$ with $\sigma \in \Sigma^{(k)}$, $\xi_1, \dots, \xi_k \in T_\Sigma$, and $\zeta_i \in C_\Sigma$. We decompose the run on ζ into two runs $\kappa_1 \in R^{\kappa'(i)}(\zeta_i)$ and $\kappa_2 \in R^{q'}(\zeta[z]_i)$ such that κ_2 maps the position of node z within $\zeta[z]_i$ to state $\kappa'(i)$. Thus, we have

$$\begin{aligned}
\langle \kappa \cdot \kappa' \rangle &= \langle \kappa \cdot (\kappa_1 \cdot \kappa_2) \rangle && \text{(decomposition } \kappa' = \kappa_1 \cdot \kappa_2) \\
&= \langle (\kappa \cdot \kappa_1) \cdot \kappa_2 \rangle && \text{(associativity)} \\
&= \langle \langle \kappa \cdot \kappa_1 \rangle \cdot \kappa_2 \rangle && \text{(induction hypothesis on } \kappa \cdot \kappa_1, \kappa_2) \\
&= \langle \langle \langle \kappa \rangle \cdot \kappa_1 \rangle \cdot \kappa_2 \rangle && \text{(induction hypothesis on } \kappa, \kappa_1) \\
&= \langle (\langle \kappa \rangle \cdot \kappa_1) \cdot \kappa_2 \rangle && \text{(induction hypothesis on } \langle \kappa \rangle \cdot \kappa_1, \kappa_2) \\
&= \langle \langle \kappa \rangle \cdot (\kappa_1 \cdot \kappa_2) \rangle && \text{(associativity)} \\
&= \langle \langle \kappa \rangle \cdot \kappa' \rangle && \text{(composition } \kappa_1 \cdot \kappa_2 = \kappa')
\end{aligned}$$

■

Proof of Observation 3.9

Proof. Let $q' \in Q$ an arbitrary state and ξ, ζ as in the observation. By definition the following statement holds: $[\![\xi \cdot \zeta]\!]_{q'} = \sum_{\kappa'' \in R^{q'}(\xi \cdot \zeta)} \langle \kappa'' \rangle$. Let $w_z \in \mathrm{pos}(\zeta)$ the position of node z within ζ. Let $\kappa''(w_z) = q$, with $q \in Q$.

Now we decompose any run κ'' into two smaller runs $\kappa \in R^q(\xi)$ and $\kappa' \in R^{q'}(\zeta)$ that coincide with κ'' on ξ and ζ, respectively:

$$\begin{aligned}
&\sum_{\kappa'' \in R^{q'}(\xi \cdot \zeta)} \langle \kappa'' \rangle \\
&= \sum_{\substack{\kappa, \kappa' \\ \kappa \cdot \kappa' \in R^{q'}(\xi \cdot \zeta)}} \langle \kappa \cdot \kappa' \rangle && \text{(decomposition)} \\
&= \sum_{\substack{\kappa, \kappa' \\ \kappa \cdot \kappa' \in R^{q'}(\xi \cdot \zeta)}} \langle \langle \kappa \rangle \cdot \kappa' \rangle && \text{(Observation 3.8)} \\
&= \sum_{\substack{\kappa' \\ [\![\xi]\!] \cdot \kappa' \in R^{q'}([\![\xi]\!] \cdot \zeta)}} \langle [\![\xi]\!] \cdot \kappa' \rangle \\
&= [\![[\![\xi]\!] \cdot \zeta]\!]_{q'}
\end{aligned}$$

Since q' was chosen arbitrarily the statement $[\![\xi \cdot \zeta]\!]_{q'} = [\![[\![\xi]\!] \cdot \zeta]\!]_{q'}$ holds for every $q \in Q$ and hence, $[\![\xi \cdot \zeta]\!] = [\![[\![\xi]\!] \cdot \zeta]\!]$. ∎

Proof of Observation 4.3

Proof. By induction on ζ. We show $[\![(u+v)\cdot\zeta]\!]_q = [\![u\cdot\zeta]\!] + [\![v\cdot\zeta]\!]_q$ for every $q \in Q$. For the induction base let $\zeta = z$:

$$[\![(u+v)\cdot z]\!]_q = (u+v)_q = u_q + v_q = [\![u]\!]_q + [\![v]\!]_q = [\![u\cdot z]\!]_q + [\![v\cdot z]\!]_q.$$

For the induction step we have to distinguish the following two cases:

(i) $\zeta = \sigma(\xi_1, \dots, \xi_{i-1}, z, \xi_{i+1}, \dots, \xi_k)$ where $\sigma \in \Sigma^{(k)}$ and $\xi_1, \dots, \xi_k \in T_\Sigma$:

$$\begin{aligned}
&[\![(u+v)\cdot\zeta]\!]_q \\
&= \sum_{\kappa \in R^q((u+v)\cdot\zeta)} \langle\kappa\rangle \\
&= \sum_{\kappa \in R^q((u+v)\cdot\zeta)} \langle\kappa|_1\rangle \cdot \ldots (u+v)_{\kappa(i)} \cdot \langle\kappa|_k\rangle \cdot \delta(\kappa(1)\ldots\kappa(i)\ldots\kappa(k), \sigma, q) \\
&= \sum_{\kappa \in R^q((u+v)\cdot\zeta)} \langle\kappa|_1\rangle \cdot \ldots (u_{\kappa(i)} + v_{\kappa(i)}) \cdot \langle\kappa|_k\rangle \cdot \delta(\kappa(1)\ldots\kappa(i)\ldots\kappa(k), \sigma, q) \\
&= \sum_{\kappa \in R^q(u\cdot\zeta)} \langle\kappa|_1\rangle \cdot \ldots u_{\kappa(i)} \cdot \langle\kappa|_k\rangle \cdot \delta(\kappa(1)\ldots\kappa(i)\ldots\kappa(k), \sigma, q) \\
&\qquad + \sum_{\kappa \in R^q(v\cdot\zeta)} \langle\kappa|_1\rangle \cdot \ldots v_{\kappa(i)} \cdot \langle\kappa|_k\rangle \cdot \delta(\kappa(1)\ldots\kappa(i)\ldots\kappa(k), \sigma, q) \\
&= [\![u\cdot\zeta]\!]_q + [\![v\cdot\zeta]\!]_q
\end{aligned}$$

(ii) $\zeta = \sigma(\xi_1, \dots, \xi_{i-1}, \zeta_i, \xi_{i+1}, \dots, \xi_k)$ with $\sigma \in \Sigma^{(k)}$, $\xi_1, \dots, \xi_k \in T_\Sigma$, and $\zeta_i \in C_\Sigma$. Hence, we decompose ζ into two smaller contexts ζ_i, ψ such that $\zeta = \zeta_i \cdot \psi$ and $\psi = \sigma(\xi_1, \dots, \xi_{i-1}, z, \xi_{i+1}, \dots, \xi_k)$.

$$\begin{aligned}
[\![(u+v)\cdot\zeta]\!]_q &= [\![(u+v)\cdot(\zeta_i\cdot\psi)]\!]_q && \text{(decomposition } \zeta = \zeta_i\cdot\psi) \\
&= [\![((u+v)\cdot\zeta_i)\cdot\psi]\!]_q && \text{(associativity)} \\
&= [\![[\![(u+v)\cdot\zeta_i)]\!]\cdot\psi]\!]_q && \text{(Observation 3.9)} \\
&= [\![([\![u\cdot\zeta_i]\!] + [\![v\cdot\zeta_i]\!])\cdot\psi]\!]_q && \text{(induction hypothesis on } \zeta_i) \\
&= [\![[\![u\cdot\zeta_i]\!]\cdot\psi]\!]_q + [\![[\![v\cdot\zeta_i]\!]\cdot\psi]\!]_q && \text{(induction hypothesis on } \psi) \\
&= [\![(u\cdot\zeta_i)\cdot\psi]\!]_q + [\![(v\cdot\zeta_i)\cdot\psi]\!]_q && \text{(Observation 3.9)} \\
&= [\![u\cdot(\zeta_i\cdot\psi)]\!]_q + [\![v\cdot(\zeta_i\cdot\psi)]\!]_q && \text{(associativity)} \\
&= [\![u\cdot\zeta]\!]_q + [\![v\cdot\zeta]\!]_q && \text{(composition } \zeta_i\cdot\psi = \zeta)
\end{aligned}$$

■

Proof of Observation 4.4

Proof. The observation follows directly from the statement below. Let $\mathcal{A}$ be a wta and $u, v \in S^{Q \cup \bar{Q}}$ arbitrary vectors. Then, for every $\zeta \in C_\Sigma$, we have

$$[\![u|_Q \cdot \zeta]\!]_{\mathcal{A}} = [\![u \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_Q \text{ and } [\![v|_{\bar{Q}} \cdot \zeta]\!]_{\mathcal{A}} = [\![v \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_{\bar{Q}}, \tag{†}$$

where for every $a \in S^{Q \cup \bar{Q}}$, i. e., $a = (a_{q_1}, \dots, a_{q_{|Q|}}, a_{\bar{q}_1}, \dots, a_{\bar{q}_{|Q|}})$, we define vectors $a|_Q, a|_{\bar{Q}} \in S^Q$ such that $a|_Q = (a_{q_1}, \dots, a_{q_{|Q|}})$ and $a|_{\bar{Q}} = (a_{\bar{q}_1}, \dots, a_{\bar{q}_{|Q|}})$.

We prove the above statement by induction on ζ. Let $\zeta = z$ for the induction base. Thus,

$$\begin{aligned}
&[\![u|_Q \cdot z]\!]_{\mathcal{A}} = [\![u|_Q]\!]_{\mathcal{A}} = u|_Q = [\![u]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_Q = [\![u \cdot z]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_Q \text{ and} \\
&[\![v|_{\bar{Q}} \cdot z]\!]_{\mathcal{A}} = [\![v|_{\bar{Q}}]\!]_{\mathcal{A}} = v|_{\bar{Q}} = [\![v]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_{\bar{Q}} = [\![v \cdot z]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_{\bar{Q}}.
\end{aligned}$$

For the induction step we dinstinguish two cases.

(i) $\zeta = \sigma(\xi_1, \dots, \xi_{i-1}, z, \xi_{i+1}, \dots, \xi_k)$ where $\sigma \in \Sigma^{(k)}$ and $\xi_1, \dots, \xi_k \in T_\Sigma$. Let $r \in Q$ an arbitrary state of $\mathcal{A}$. Then, we have

$$\begin{aligned}
&([\![u|_Q \cdot \zeta]\!]_{\mathcal{A}})_r \\
&= ([\![\sigma(\xi_1, \dots, \xi_{i-1}, u|_Q, \xi_{i+1}, \dots, \xi_k)]\!]_{\mathcal{A}})_r && \text{(concatenation)} \\
&= \sum_{\kappa \in R^r(u|_Q \cdot \zeta)} \langle \kappa \rangle_{\mathcal{A}} && \text{(by definition)} \\
&= \sum_{\kappa \in R^r(u|_Q \cdot \zeta)} \langle \kappa|_1 \rangle_{\mathcal{A}} \cdot \ldots \cdot \langle \kappa|_i \rangle_{\mathcal{A}} \cdot \ldots \cdot \langle \kappa|_k \rangle_{\mathcal{A}} \cdot \delta(\kappa(1) \dots \kappa(i) \dots \kappa(k), \sigma, r) \\
& && \text{(by definition)} \\
&= \sum_{\kappa \in R^r(u \cdot \zeta)} \langle \kappa|_1 \rangle_{\mathcal{A} \cup \bar{\mathcal{A}}} \cdot \ldots \cdot \langle \kappa|_i \rangle_{\mathcal{A} \cup \bar{\mathcal{A}}} \cdot \ldots \cdot \langle \kappa|_k \rangle_{\mathcal{A} \cup \bar{\mathcal{A}}} \cdot \delta'(\kappa(1) \dots \kappa(i) \dots \kappa(k), \sigma, r) \\
& && (*) \\
&= \sum_{\kappa \in R^r(u \cdot \zeta)} \langle \kappa \rangle_{\mathcal{A} \cup \bar{\mathcal{A}}} && \text{(by definition)} \\
&= ([\![\sigma(\xi_1, \dots, \xi_{i-1}, u, \xi_{i+1}, \dots, \xi_k)]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_Q)_r && \text{(by definition)} \\
&= ([\![u \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_Q)_r
\end{aligned}$$

At $(*)$ we used the following facts:

- The wta $\mathcal{A}$ assigns a weight $\langle \kappa|_i \rangle_{\mathcal{A}}$ the the according run on $u|_Q$. Using the induction base, we conclude that the union wta $\mathcal{A} \cup \bar{\mathcal{A}}$ assigns the weight $\langle \kappa|_i \rangle_{\mathcal{A} \cup \bar{\mathcal{A}}}$.
- Any run κ in the sum decorates the nodes of the given tree with the same states, independent of whether the weight $\langle \kappa \rangle$ is assigned by $\mathcal{A}$ or $\mathcal{A} \cup \bar{\mathcal{A}}$. Theoretically, a run on ζ that is evaluated by the union wta could decorate

nodes with states $\bar{r} \in \bar{Q}$ in addition to simply using states from Q. However, if this happens, the weights of transitions over states from both Q and $\bar{Q}$ are zero. Hence, they do not contribute to the final sum. Transitions over states that are only taken from $\bar{Q}$ would have a non-zero weight based on $\bar{\delta}$. However, since the root of ζ is decorated with some $r \in Q$ there would have to be at least one transistion over mixed states from both Q and $\bar{Q}$. Thus, $\langle \kappa|_\ell \rangle_{\mathcal{A}} = \langle \kappa|_\ell \rangle_{\mathcal{A} \cup \bar{\mathcal{A}}}$ for every $\ell \in \{1, \ldots, k\} \setminus \{i\}$.

- The weights of transitions δ' and δ coincide. By similar reasoning as above, we do not need to consider weights of transistions over states from both Q and $\bar{Q}$. We are only interested in the weight vector $[\![u \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_Q$ and on those stated both transitions and their weights coincide.

As r was chosen arbitrarily, the statement $([\![u|_Q \cdot \zeta]\!]_{\mathcal{A}})_r = ([\![u \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_Q)_r$ holds for every state $r \in Q$, and, thus, $[\![u|_Q \cdot \zeta]\!]_{\mathcal{A}} = [\![u \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_Q$. Similar considerations lead to $[\![v|_{\bar{Q}} \cdot \zeta]\!]_{\mathcal{A}} = [\![v \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_{\bar{Q}}$.

(ii) $\zeta = \sigma(\xi_1, \ldots, \xi_{i-1}, \zeta_i, \xi_{i+1}, \ldots, \xi_k)$ with $\sigma \in \Sigma^{(k)}$, $\xi_1, \ldots, \xi_k \in T_\Sigma$, and $\zeta_i \in C_\Sigma$. Hence, we decompose ζ into two smaller contexts ζ_i, ψ such that $\zeta = \zeta_i \cdot \psi$ and $\psi = \sigma(\xi_1, \ldots, \xi_{i-1}, z, \xi_{i+1}, \ldots, \xi_k)$.

$$\begin{aligned}
[\![u|_Q \cdot (\zeta_i \cdot \psi)]\!]_{\mathcal{A}} &= [\![(u|_Q \cdot \zeta_i) \cdot \psi]\!]_{\mathcal{A}} && \text{(associativity)} \\
&= [\![[\![u|_Q \cdot \zeta_i]\!]_{\mathcal{A}} \cdot \psi]\!]_{\mathcal{A}} && \text{(Observation 3.9)} \\
&= [\![[\![u \cdot \zeta_i]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_Q \cdot \psi]\!]_{\mathcal{A}} && \text{(induction hypothesis on } \zeta_i) \\
&= [\![[\![u \cdot \zeta_i]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}} \cdot \psi]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_Q && \text{(induction hypothesis on } \psi) \\
&= [\![(u \cdot \zeta_i) \cdot \psi]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_Q && \text{(Observation 3.9)} \\
&= [\![u \cdot (\zeta_i \cdot \psi)]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_Q && \text{(associativity)}
\end{aligned}$$

We obtain $[\![v|_{\bar{Q}} \cdot (\zeta_i \cdot \psi)]\!]_{\mathcal{A}} = [\![v \cdot (\zeta_i \cdot \psi)]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_{\bar{Q}}$ in a similar way.

Using (†), we derive

$$\begin{aligned}
&([\![e_p \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}})_p = ([\![e_p \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_Q)_p = ([\![e_p|_Q \cdot \zeta]\!]_{\mathcal{A}})_p \text{ and} \\
&([\![e_{\bar{q}} \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}})_{\bar{q}} = ([\![e_{\bar{q}} \cdot \zeta]\!]_{\mathcal{A} \cup \bar{\mathcal{A}}}|_{\bar{Q}})_q = ([\![e_{\bar{q}}|_{\bar{Q}} \cdot \zeta]\!]_{\mathcal{A}})_q,
\end{aligned}$$

where $e_p|_Q, e_{\bar{q}}|_{\bar{Q}} \in S^Q$ are the corresponding e-vectors for $\mathcal{A}$. Hence, our observation stating $(p, q) \in \textsc{Twins}(\mathcal{A}) \Leftrightarrow (p, \bar{q}) \in \textsc{Twins}(\mathcal{A} \cup \bar{\mathcal{A}})$ follows directly from the above equations. ∎

Proof of Observation 4.7

Proof. Let $(p, q) \in \textsc{Siblings}(\mathcal{A})$ and $u \in S^{Q \cup \bar{Q}}$ a critical vector with $u \neq \tilde{0}$. Thus, we have that $u_p \neq u_{\bar{q}}$. Moreover, we know that $u = g(u) \cdot f(u)$. Hence, we derive

$$\begin{aligned}
& u_p \neq u_{\bar{q}} \\
\Leftrightarrow \quad & g(u) \cdot (f(u))_p \neq g(u) \cdot (f(u))_{\bar{q}} \\
\Leftrightarrow \quad & g^{-1}(u) \cdot g(u) \cdot (f(u))_p \neq g^{-1}(u) \cdot g(u) \cdot (f(u))_{\bar{q}} \\
\Leftrightarrow \quad & (f(u))_p \neq (f(u))_{\bar{q}}
\end{aligned}$$

■

Proof of Observation 4.14

Proof. Let $X \subseteq M$ a nonempty countable chain and (T_0, C_0), $(T_1, C_1) \in X$ two arbitrary elements of X such that $(T_0, C_0) \sqsubseteq (T_1, C_1)$. We compute $(T_0', C_0') := F(T_0, C_0)$ and $(T_1', C_1') := F(T_1, C_1)$, and show that $(T_0', C_0') \sqsubseteq (T_1', C_1')$:

(i) Let u an arbitrary element of T_0'. It had to be constructed by making use of $(F1)$. Thus, there are $k \geq 0$, $\sigma \in \Sigma^{(k)}$, and $u_1, \ldots, u_k \in T_0$ such that $\llbracket \sigma(u_1, \ldots, u_k) \rrbracket \neq \tilde{0}$. Hence, $u = f\llbracket \sigma(u_1, \ldots, u_k) \rrbracket$. Moreover, we have that $u_1, \ldots, u_k \in T_1$ because $T_0 \subseteq T_1$. By applying $(F1)$ to (T_1, C_1) we obtain $f\llbracket \sigma(u_1, \ldots, u_k) \rrbracket \in T_1'$ and, thus, $u \in T_1'$. Since $u \in T_0'$ was chosen arbitrarily, we conclude $T_0' \subseteq T_1'$.

(ii) Let u an arbitrary element of C_0'. We have to distinguish two different cases:

(a) $u = f(e_p + e_{\bar{q}})$. Then, by definition of $(F2)$ we have that u is also in C_1'.

(b) $u = f\llbracket \sigma(u_1, \ldots, u_k) \rrbracket$. Then, u has been produced by applying $(F3)$ to (T_0, C_0). Hence, there exist $k \geq 0$, $\sigma \in \Sigma^{(k)}$, $i \in \{1, \ldots, k\}$, $u_i \in C_0$, and $u_\ell \in T_0$ for every $\ell \in \{1, \ldots, k\} \setminus \{i\}$ such that $\llbracket \sigma(u_1, \ldots, u_k) \rrbracket \neq \tilde{0}$. By $T_0 \subseteq T_1$ and $C_0 \subseteq C_1$ we obtain $u_\ell \in T_1$ for every $\ell \in \{1, \ldots, k\} \setminus \{i\}$ and $u_i \in C_1$. Applying $(F3)$ to (T_1, C_1) leads to $u = f\llbracket \sigma(u_1, \ldots, u_k) \rrbracket \in C_1'$.

As a result of choosing $u \in C_0'$ arbitrarily we get $C_0' \subseteq C_1'$.

By the definition of $\sqsubseteq$, we obtain $(T_0', C_0') \sqsubseteq (T_1', C_1')$. ■

Proof of Observation 4.15

Proof. Let $X = \{(T_m, C_m) \mid m \geq 0\} \subseteq M$ an arbitrary nonempty countable chain. The supremum of X exists due to the ω-completeness of M. By Observation 4.14 we have that $(T_m, C_m) \sqsubseteq (T_n, C_n)$ implies $F(T_m, C_m) \sqsubseteq F(T_n, C_n)$ for arbitrary two elements $(T_m, C_m), (T_n, C_n) \in X$, with $m, n \geq 0$. Thus, $F(X)$ itself is a countable chain in M, and by the ω-completeness of M the supremum of $F(X)$ exists.

Now we show $F(\sup X) = \sup F(X)$: First, we compute both terms. Note that whenever we apply F to an element $(T, C) \in M$ we abbreviate the output with (T', C'). Thus, we obtain

$$\begin{aligned}
(T'_A, C'_A) &:= F(\sup X) \\
&= F(\sup \{T_m \mid m \geq 0\}, \sup \{C_m \mid m \geq 0\}) = F(\overbrace{\bigcup_{m\geq 0} T_m}^{T_A}, \overbrace{\bigcup_{m\geq 0} C_m}^{C_A}) \text{ and} \\
(T'_B, C'_B) &:= \sup F(X) \\
&= (\sup \{T'_i \mid i \geq 0\}, \sup \{C'_i \mid i \geq 0\}) = (\bigcup_{m\geq 0} T'_m, \bigcup_{m\geq 0} C'_m).
\end{aligned}$$

Hence, we show $(T'_A, C'_A) = (T'_B, C'_B)$ by proving the following two directions.

(i) "$\sqsubseteq$": Let $u \in T'_A$ arbitrary. By $(F1)$ there are $k \geq 0$, $\sigma \in \Sigma^{(k)}$, $u_1, \ldots, u_k \in T_A$ such that $[\![\sigma(u_1, \ldots, u_k)]\!] \neq \tilde{0}$, and, hence, $u = f[\![\sigma(u_1, \ldots, u_k)]\!]$. From $u_1, \ldots, u_k \in T_A$ we conclude that every u_i with $i \in \{1, \ldots, k\}$ has to be an element of some T_m, $m \geq 0$. However, since the sets T_m are first components of a countable chain, there has to be an element $(T_x, C_x) \in X$, $x \geq 0$ such that T_x contains all $u_1, \ldots, u_k$. Then, by applying $(F1)$ to (T_x, C_x) we obtain $u = f[\![\sigma(u_1, \ldots, u_k)]\!] \in T'_x$. Hence, u is also in T'_B.

Let $u \in C'_A$ arbitrary. Then, either $u = f(e_p + e_{\bar{q}})$ for some $(p, q) \in \textsc{Siblings}(\mathcal{A})$, which, by definition of $(F2)$, leads to $u \in C'_B$, or $u = f[\![\sigma(u_1, \ldots, u_k)]\!]$. In the latter case we have by $(F3)$ that there are $k \geq 0$, $\sigma \in \Sigma^{(k)}$, $i \in \{1, \ldots, k\}$, $u_i \in C_A$, and $u_\ell \in T_A$ for every $\ell \in \{1, \ldots, k\} \setminus \{i\}$ such that $[\![\sigma(u_1, \ldots, u_k)]\!] \neq \tilde{0}$. By the same argumentation as above there is an element $(T_x, C_x) \in X$, $x \geq 0$ such that T_x contains all $u_\ell k$, for every $\ell \in \{1, \ldots, k\} \setminus \{i\}$ and C_x contains u_i. Hence, $u \in C'_x$ and also $u \in C'_B$.

Because we chose elements of T'_A and C'_A arbitrarily we obtain $T'_A \subseteq T'_B$ and $C'_A \subseteq C'_B$. Hence, $(T'_A, C'_A) \sqsubseteq (T'_B, C'_B)$.

(ii) "$\sqsupseteq$": Let $u \in T'_B$ arbitrary. By the definition of T'_B we know that $u \in T'_m$ for some $m \geq 0$. By $(F1)$ there exist $k \geq 0$, $\sigma \in \Sigma^{(k)}$, and $u_1, \ldots, u_k \in T_m$ such that $[\![\sigma(u_1, \ldots, u_k)]\!] \neq \tilde{0}$ which leads to $u = f[\![\sigma(u_1, \ldots, u_k)]\!]$. Since $u_1, \ldots, u_k \in T_m$ we also have that $u_1, \ldots, u_k \in T_A$. Thus, by $(F1)$ on (T_A, C_A) we get $u \in T'_A$.

Let $u \in C'_B$ arbitrary. Then, either $u = f(e_p + e_{\bar{q}})$ for some $(p, q) \in \textsc{Siblings}(\mathcal{A})$, which, by definition of $(F2)$, leads to $u \in C'_A$, or $u = f[\![\sigma(u_1, \ldots, u_k)]\!]$. In the latter case we have that $u \in C'_m$ for some $m \geq 0$ and by $(F3)$ there are $k \geq 0$, $\sigma \in \Sigma^{(k)}$, $i \in \{1, \ldots, k\}$, $u_i \in C_m$, and $u_\ell \in T_m$ for every $\ell \in \{1, \ldots, k\} \setminus \{i\}$ such that $[\![\sigma(u_1, \ldots, u_k)]\!] \neq \tilde{0}$. Hence, $u_i \in C_A$ and $u_\ell \in T_A$ for every $\ell \in \{1, \ldots, k\} \setminus \{i\}$. As a result of applying $(F3)$ to (T_A, C_A) we have that $u \in C'_A$.

Since elements of T'_B and C'_B were chosen arbitrarily we obtain $T'_B \subseteq T'_A$ and $C'_B \subseteq C'_A$. Hence, $(T'_B, C'_B) \sqsubseteq (T'_A, C'_A)$. ∎

Bibliography

[AM03] Cyril Allauzen and Mehryar Mohri. Efficient algorithms for testing the twins property. *J. Autom. Lang. Comb.*, 8:117–144, April 2003.

[BMV10] Matthias Büchse, Jonathan May, and Heiko Vogler. Determinization of weighted tree automata using factorizations. *Journal of Automata, Languages and Combinatorics*, 15(3/4), 2010.

[Bor04] Björn Borchardt. A pumping lemma and decidability problems for recognizable tree series. *Acta Cybern.*, 16:509–544, September 2004.

[BV03] Björn Borchardt and Heiko Vogler. Determinization of finite state weighted tree automata. *J. Autom. Lang. Combin*, 8, 2003.

[CdlH00] Francisco Casacuberta and Colin de la Higuera. Computational complexity of problems on probabilistic grammars and transducers. In *Proc. ICGI*, pages 15–24. Springer, 2000.

[Cho77] Christian Choffrut. Une caracterisation des fonctions sequentielles et des fonctions sous-sequentielles en tant que relations rationnelles. *Theor. Comput. Sci.*, 5(3):325–337, 1977.

[ÉK03] Zoltán Ésik and Werner Kuich. Formal tree series. *J. Autom. Lang. Comb.*, 8(2), 2003.

[Epp98] David Eppstein. Finding the k shortest paths. *SIAM Journal on Computing*, 28(2), 1998.

[FV09] Zoltán Fülöp and Heiko Vogler. Weighted tree automata and tree transducers. In *Handbook of Weighted Automata*, chapter 9. Springer, 2009.

[Gol99] Jonathan Samuel Golan. *Semirings and their Applications*. Kluwer Academic Publishers, 1999.

[GS84] Ferenc Gécseg and Magnus Steinby. *Tree Automata*. Akadéniai Kiadó, Budapest, Hungary, 1984.

[GTWW77] Joseph A. Goguen, James W. Thatcher, Eric G. Wagner, and Jesse B. Wright. Initial algebra semantics and continuous algebras. *J. ACM*, 24:68–95, January 1977.

[HC05] Liang Huang and David Chiang. Better k-best parsing. In *Proc. IWPT*, pages 53–64. ACL, 2005.

[HW98] Udo Hebisch and Hanns Joachim Weinert. *Semirings: Algebraic Theory and Applications in Computer Science*, volume 5 of *Series in Algebra*. World Scientific, 1998.

[Kir12] Daniel Kirsten. Decidability, undecidability, and pspace-completeness of the twins property in the tropical semiring. *Theor. Comput. Sci.*, 420:56–63, 2012.

[KM05] Daniel Kirsten and Ina Mäurer. On the determinization of weighted automata. *J. Autom. Lang. Comb.*, 10:287–312, 2005.

[Mah84] Bernd Mahr. Iteration and summability in semirings. *Annals of Discrete Mathematics*, 19:229–256, 1984.

[Moh97] Mehryar Mohri. Finite-state transducers in language and speech processing. *Computational Linguistics*, 23(2):269–311, 1997.

[Sim96] Khalil Sima'an. Computational complexity of probabilistic disambiguation by means of tree-grammars. In *Proc. COLING*, pages 1175–1180, 1996.

[Wec92] Wolfgang Wechler. *Universal Algebra for Computer Scientists*, volume 25 of *Monogr. Theoret. Comput. Sci. EATCS Ser.* Springer, 1992.