# A Kleene theorem for weighted tree automata


Manfred Droste
*Department of Mathematics and Natural Sciences*
E-mail: `droste@math.tu-dresden.de`

Heiko Vogler
*Department of Computer Science*
E-mail: `vogler@inf.tu-dresden.de`

*Dresden University of Technology*
*D-01062 Dresden, Germany*



May 24, 2002



**Abstract:** In this paper we prove Kleene's result for tree series over a commutative and idempotent semiring $A$ (which is not necessarily complete or continuous), i.e., the class of recognizable tree series over $A$ and the class of rational tree series over $A$ are equal. We show the result by direct automata-theoretic constructions and prove their correctness.


## 1 Introduction

In automata theory, Kleene's fundamental theorem [Kle56] on the coincidence of regular and rational languages has been extended in several directions. Thatcher and Wright [TW68] generalized it to tree languages, also cf. [GS84, Eng75b, ES77]. Schützenberger [Sch61] showed that the formal power series (cost functions) associated with weighted finite automata over words and an arbitrary semiring for the weights, are precisely the rational formal power series. Weighted automata have recently received much interest due to their applications in image compression (Culik II and Kari [CK93], Hafner [Haf99], Katritzke [Kat01], Jiang, Litow and de Vel [JLdV00]) and in speech-to-text processing (Mohri [Moh97], [MPR00], Buchsbaum, Giancarlo and Westbrook [BGW00]). For theoretical background on formal power series, we refer the reader to [SS78, KS86, BR88, Kui97b].

In this paper, we wish to extend Thatcher and Wright's and Schützenberger's approaches to weighted automata on trees, for short: weighted tree automata. In the case where the weight semiring is a field, Berstel and Reutenauer [BR82] showed that the recognizable series again coincide with the rational ones. Kuich [Kui97a, Kui99] established the analogous result for ordered commutative semirings which are complete and continuous, i.e., infinite sums exist and satisfy an infinitary distributivity law. Here, we will investigate weighted tree automata where the semiring is idempotent and commutative. Such semirings are naturally ordered but no completeness assumptions are made (leading to a much easier (first order) axiomatization). Idempotent semirings exist in abundance (e.g. any subsemigroup of the max-plus semiring of the real numbers), are fundamental in max-plus algebra (Gaubert and Plus [GP97, Gau01], Cuninghame-Green [CG95]) and in algebraic optimization problems ([Zim81]), and also occurred in other investigations on formal power series (e.g., [Kui97b, DG99, DG00]).

In [Sei92, Sei94] weigthed tree automata were introduced and decidability results for particular semirings over $\mathbb{N}$ were proved. In [FSW94] weighted tree automata were used for efficient code selection. If the weighted tree automata also produce output trees, then tree series transducers are obtained [Kui98, EFV02]; they generalize the concept of tree transducers [Rou70, Tha70, Eng75a, Bak79].

We will consider the concept of an automaton acting on (finite) trees as usual, but with weights (costs) for the transitions. Their behaviour is described by the function associating to each tree the cost the automaton needs for evaluating it. The classical tree language theoretic case is obtained when the weights take values in the Boolean semiring. Then we describe rational operations like sum, top-concatenation, $\alpha$-product and two kinds of $\alpha$-iteration generalizing naturally the classical ones for tree languages investigated by Thatcher and Wright [TW68] and by Engelfriet [Eng75b]. We show that these two kinds of $\alpha$-iterations actually coincide for commutative semirings, leading to an $\alpha$-Kleene star operation for tree series. We also compare our operations with the corresponding ones for series over words. *Rational series* then are those which can be obtained by these operations from the monomials (or, equivalently, from the polynomials).

Our main result states that for any idempotent and commutative semiring $A$, the recognizable and the rational formal tree series over $A$ coincide. As in the tree language theoretic case, here one has to be careful about the underlying alphabets involved. We show that any rational tree series is recognizable with automata acting on the same ranked alphabet, but a tree series associated to a finite automaton is rational where now the underlying alphabet is extended by the state set of the finite automaton. Thus, our main result is the following:

$$\text{Let } A \text{ be a commutative and idempotent semiring. Then}$$
$$A^{rec}\langle\langle T_\Sigma \rangle\rangle \subseteq \bigcup_{Q \text{ finite set}} A^{rat}\langle\langle T_\Sigma(Q) \rangle\rangle$$
$$A^{rat}\langle\langle T_\Sigma \rangle\rangle \subseteq A^{rec}\langle\langle T_\Sigma \rangle\rangle.$$

where $A^{rec}\langle\langle T_\Sigma \rangle\rangle$ and $A^{rat}\langle\langle T_\Sigma \rangle\rangle$ (and $A^{rat}\langle\langle T_\Sigma(Q) \rangle\rangle$) are the classes of recognizable and rational, resp. formal tree series over $A$ and $T_\Sigma$ (and $T_\Sigma(Q)$) (cf. Theorems 4.1 and 5.10).

We obtain the classical case of tree languages, i.e. the result of Thatcher and Wright, as an immediate consequence by taking the weights in the Boolean semiring. By restricting the tree alphabets to string alphabets, we also obtain Schützenberger's result on formal power series over words (but here only for idempotent semirings). For the proofs of our results, we give the explicit constructions of the weighted tree automata involved. These constructions are *effective* in the usual sense, leading from a rational expression to a weighted tree automaton and conversely. The correctness arguments for the constructions then require a detailed analysis of the fine structure of the trees occurring.

In Section 2 we recall some notions of semirings, formal power series, trees, tree languages, formal tree series and operations on each of these objects. Also, we define the usual rational operations on formal tree series. The reader who is familiar with these concepts can skip over the preliminaries on first reading. In Section 3 we introduce bottom-up weighted tree automata and show a normal form result. In Sections 4 and 5 we prove the two directions of Kleene's result. Finally, in Section 6 we compare our result with the other Kleene-results for formal tree series which occur in the literature.

# 2 Preliminaries

## 2.1 Semirings and formal power series

For a survey paper about the relevance of semirings and formal power series to formal languages and automata cf. [Kui97b] (also cf. [KS86]). Here we only recall the needed definitions.

A *semiring* is an algebraic structure $(A, \oplus, \odot, 0, 1)$ with two operations sum $\oplus$ and product $\odot$ such that $(A, \oplus, 0)$ is a commutative monoid, $(A, \odot, 1)$ is a monoid, and for every $a, b, c \in A$ the following distributivity laws hold: $(a \oplus b) \odot c = a \odot c \oplus b \odot c$, $a \odot (b \oplus c) = a \odot b \oplus a \odot c$ where we assume that $\odot$ has higher binding priority than $\oplus$, and $a \odot 0 = 0 \odot a = 0$. Whenever the operations ($\oplus$ and $\odot$) and the neutral elements (0 and 1) are clear from the context, then we denote the semiring $(A, \oplus, \odot, 0, 1)$ just by $A$. As usual, we define $\sum_{i\in\emptyset} a_i = 0$ and $\prod_{i\in\emptyset} a_i = 1$. A semiring $A$ is *idempotent* if $a \oplus a = a$ for every $a \in A$. Next we list some of the most common semirings:

- the *Boolean semiring* $\mathbb{B} = (\{0, 1\}, \vee, \wedge, 0, 1)$ with disjunction and conjunction as sum and product, resp. and 0 and 1 as $false$ and $true$, resp.; this semiring links the theory of formal power series and the theory of formal languages;
- the *semiring of natural numbers (with $\infty$)* $Nat_\infty = (\mathbb{N} \cup \{\infty\}, +, \cdot, 0, 1)$ with the obvious addition and multiplication operations;
- the *tropical semiring* $Trop = (\mathbb{N} \cup \{\infty\}, min, +, \infty, 0)$ and the operations $min$ and $+$ are extended to $\mathbb{N} \cup \{\infty\}$ in the obvious way;
- $\mathbb{Q}_+ = (\mathbb{Q} \cup \{-\infty\}, max, +, -\infty, 0)$;
- $A = ([0, 1]_{\mathbb{Q}}, max, \cdot, 0, 1, )$ which can be used to compute probabilities.

In the rest of this paper $A$ denotes an arbitrary semiring $(A, \oplus, \odot, 0, 1)$.

Let $S$ be a set. A *formal power series (over $S$ and $A$)* is a mapping $\varphi : S \to A$. For every $s \in S$, the element $\varphi(s) \in A$ is called *coefficient* of $s$ and it is also denoted by $(\varphi, s)$; then the mapping $\varphi$ itself is written as the formal sum $\sum_{s\in S}(\varphi, s)\, s$. Given two series $\varphi, \psi : S \to A$, we define the *sum* $\varphi \oplus \psi : S \to A$ by pointwise addition, that is, $(\varphi \oplus \psi, s) = (\varphi, s) \oplus (\psi, s)$ for every $s \in S$.

Let $\varphi : S \to A$ be a formal power series. For every $a \in A$ we define $a \odot \varphi$ by $(a \odot \varphi, s) = a \odot (\varphi, s)$ for every $s \in S$. The *support of* $\varphi$ is defined as the set $supp(\varphi) = \{s \in S \,|\, (\varphi, s) \neq 0\}$. Moreover, the tree series $\varphi$ is called a *polynomial* if $supp(\varphi)$ is finite, and $\varphi$ is called a *monomial* if $supp(\varphi)$ is a singleton. A polynomial $\varphi$ is written as $a_1\, s_1 \oplus \ldots \oplus a_n\, s_n$, where $supp(\varphi) = \{s_1, \ldots, s_n\}$ and $a_i = (\varphi, s_i)$ for every $1 \leq i \leq n$. The *set of all formal power series over $S$ and $A$* (*polynomials, monomials*) is denoted by $A\langle\langle S\rangle\rangle$ (and $A\langle S\rangle$, $A[S]$, respectively). We let $\tilde{0}$ denote the formal power series with constant value 0. Clearly, $(A\langle\langle S\rangle\rangle, \oplus, \tilde{0})$ and $(A\langle S\rangle, \oplus, \tilde{0})$ are commutative monoids. We identify every element $s \in S$ with the monomial $1\, s$. Thus, $S \subseteq A[S] \subseteq A\langle S\rangle \subseteq A\langle\langle S\rangle\rangle$.

A family $(\varphi_i \,|\, i \in I)$ of formal power series $\varphi_i : S \to A$ is called *locally finite*, if for every $s \in S$, there are only finitely many $i \in I$ such that $(\varphi_i, s) \neq 0$. In this case, we define the *sum* by $\sum_{i\in I} \varphi_i = \sum_{s\in S}(\sum_{i\in I}(\varphi_i, s))s$. Then every series $\varphi$ is the sum of the monomials $(\varphi, s)\, s$ for $s \in S$.

We define the update of a formal power series as follows. Let $\varphi : S \to A$, $s \in S$, and $a \in A$. Define the formal power series $\varphi[s/a] : S \to A$ by $(\varphi[s/a], t) = a$ if $t = s$ and $(\varphi[s/a], t) = (\varphi, t)$ otherwise $(t \in S)$.

## 2.2 Strings

Let $\Delta$ be an alphabet (i.e., a non-empty finite set). Then $\Delta^*$ denotes the set of all finite sequences (or: *words*) over $\Delta$, and $\varepsilon$ denotes the *empty word*; for every $w \in \Delta^*$, $length(w)$ denotes the length, i.e., number of symbols of $w$. A subset $L \subseteq \Delta^*$ is called *formal language*. For two formal languages $K, L \subseteq \Delta^*$, the *concatenation of $K$ and $L$* is the formal language $K \cdot L = \{vw \,|\, v \in K, w \in L\}$. The *Kleene star of $L$* is the set $L^* = \bigcup_{n\geq 0} L^n$ where $L^0 = \{\varepsilon\}$ and for every $n \geq 0$, $L^{n+1} = L^n \cdot L$.

Let $\varphi, \psi \in A\langle\langle\Delta^*\rangle\rangle$ be two formal power series over $\Delta^*$. (Recall that $A$ denotes an arbitrary semiring.) The *Cauchy product of $\varphi$ and $\psi$*, denoted by $\varphi \cdot_C \psi$ is the formal power series such that for every $w \in \Delta^*$, $(\varphi \cdot_C \psi, w) = \sum_{w=uv}(\varphi, u) \odot (\psi, v)$ (cf. [BR88, Kui97b]). We let the Cauchy product have weaker (stronger, resp.) binding priority than the update parentheses (the semiring operations $\oplus$ and $\odot$, resp.), i.e., $\varphi \cdot_C \psi[w/a] = \varphi \cdot_C (\psi[w/a])$ (and $\varphi \cdot_C \psi \oplus \chi = (\varphi \cdot_C \psi) \oplus \chi$, resp.). We note that the Cauchy product is associative and distributes over $\oplus$, so $(A\langle\langle\Delta^*\rangle\rangle, \oplus, \cdot_C, \tilde{0}, 1\varepsilon)$ is a semiring.

Let $R \subseteq S$. Then $R$ corresponds to the characteristic function $1_R \in \mathbb{B}\langle\langle S\rangle\rangle$ of $R$ (viewed as formal power series) such that $supp(1_R) = R$ (note that $1_R$ is uniquely determined). Vice versa, let $\varphi \in \mathbb{B}\langle\langle S\rangle\rangle$. Then $\varphi$ corresponds to the subset $R = supp(\varphi) \subseteq S$. Thus we obtain the following observation.

**Fact 2.1** The semirings $(\mathcal{P}(\Delta^*), \cup, \cdot, \emptyset, \{\varepsilon\})$ and $(\mathbb{B}\langle\langle\Delta^*\rangle\rangle, \oplus, \cdot_C, \tilde{0}, 1\varepsilon)$ are isomorphic.

Later we will need the following technical lemma.

**Lemma 2.2** Let $\varphi, \psi \in A\langle\langle\Delta^*\rangle\rangle$. Then $\varphi \cdot_C \psi[\varepsilon/1] = \varphi \cdot_C \psi[\varepsilon/0] \oplus \varphi$.

**Proof.** Let $w \in \Delta^*$. Then

$$\begin{aligned}
& (\varphi \cdot_C \psi[\varepsilon/1], w) \\
= \;& \textstyle\sum_{w=uv}(\varphi, u) \odot (\psi[\varepsilon/1], v) \\
= \;& (\varphi, w) \odot (\psi[\varepsilon/1], \varepsilon) \oplus \textstyle\sum_{\substack{w=uv\\ v\neq\varepsilon}}(\varphi, u) \odot (\psi[\varepsilon/1], v) \\
= \;& (\varphi, w) \oplus \textstyle\sum_{w=uv}(\varphi, u) \odot (\psi[\varepsilon/0], v) \\
= \;& (\varphi, w) \oplus (\varphi \cdot_C \psi[\varepsilon/0], w) \\
= \;& (\varphi \cdot_C \psi[\varepsilon/0] \oplus \varphi, w)
\end{aligned}$$

□

Let $\varphi \in A\langle\langle\Delta^*\rangle\rangle$. For every $n \geq 0$, the $n$-th Kleene-power is defined inductively by $\varphi^{0,K} = 1\varepsilon$ and for every $n \geq 0$, $\varphi^{n+1,K} = \varphi^{n,K} \cdot_C \varphi$. Note that, for every $n \geq 1$, $\varphi^{n,K} = \underbrace{\varphi \cdot_C \cdots \cdot_C \varphi}_{n \text{ factors}}$.

For every *proper* formal power series $\varphi$, i.e., $(\varphi, \varepsilon) = 0$, the *Kleene-star of* $\varphi$ is the formal power series $\varphi^{*,K} = \sum_{n \geq 0} \varphi^{n,K}$. Here the family $\varphi^{n,K}$ $(n \geq 0)$ is locally finite. Note that, for every $w \in \Delta^*$ and $n > \mathit{length}(w)$, $(\varphi^{n,K}, w) = 0$, because $(\varphi^{n,K}, w) = \sum_{w=w_1 \ldots w_n} (\varphi, w_1) \odot \ldots \odot (\varphi, w_n)$ and in every such decomposition $w_1 \ldots w_n$ of $w$ at least one of the $w_i$'s must be the empty word; then, because $\varphi$ is proper $(\varphi, w_i) = 0$. Thus we can restrict the summation of the Kleene-powers in the Kleene-star to the length of the word to which the Kleene-star is applied.

**Observation 2.3** Let $w \in \Delta^*$ and $\varphi \in A\langle\langle \Delta^* \rangle\rangle$ be a proper formal power series. Then, $(\varphi^*, w) = \sum_{n=0}^{\mathit{length}(w)} (\varphi^{n,K}, w)$.

## 2.3 Trees

A *ranked alphabet* is a tuple $(\Sigma, rk)$ where $\Sigma$ is a finite set and $rk : \Sigma \to \mathbb{N}$ is a mapping called rank mapping. For every $k \geq 0$, we define $\Sigma^{(k)} = \{\sigma \in \Sigma \mid rk(\sigma) = k\}$, the set of all symbols of $\Sigma$ which have rank $k$. Let $\alpha \in \Sigma^{(0)}$. $\Sigma$ is called an *$\alpha$-string alphabet* if $\Sigma^{(0)} = \{\alpha\}$ and $\Sigma^{(k)} = \emptyset$ for every $k \geq 2$.

In the rest of this paper let $\Sigma$ be a ranked alphabet. Moreover, let $H$ be a set disjoint with $\Sigma$. The set of *(finite, labeled and ordered) trees over* $\Sigma$ *indexed by* $H$, denoted by $T_\Sigma(H)$, is the smallest set $T \subseteq (\Sigma \cup H \cup \{(,),,\})^*$ such that (i) $H \cup \Sigma^{(0)} \subseteq T$ and (ii) if $k \geq 1$, $\sigma \in \Sigma^{(k)}$, and $s_1, \ldots, s_k \in T$, then $\sigma(s_1, \ldots, s_k) \in T$. The set $T_\Sigma(\emptyset)$ is denoted by $T_\Sigma$. Thus, the elements of $H \cup \Sigma^{(0)}$ occur only at the leaves of a tree $t \in T_\Sigma(H)$, and all leaves of such a tree $t$ belong to $H \cup \Sigma^{(0)}$. Hence, by viewing $H$ as a set of nullary symbols, we have $T_{\Sigma \cup H} = T_\Sigma(H)$.

The *size* of a tree is the number of its inner nodes, i.e., for every $t \in T_\Sigma(H)$, (i) if $t \in H \cup \Sigma^{(0)}$, then $\mathit{size}(t) = 1$ and (ii) if $t = \sigma(s_1, \ldots, s_k)$, then $\mathit{size}(t) = 1 + \sum_{i=1}^{k} \mathit{size}(s_i)$.

Let $\Sigma$ be an $\alpha$-string alphabet. Then every tree $t \in T_\Sigma$ has the form $t = \gamma_1(\gamma_2(\ldots \gamma_n(\alpha) \ldots))$ for some $\gamma_1, \ldots, \gamma_n \in \Sigma^{(1)}$. The mapping $\mathit{flat} : T_\Sigma \to (\Sigma - \{\alpha\})^*$, defined for every tree $t = \gamma_1(\gamma_2(\ldots \gamma_n(\alpha) \ldots)) \in T_\Sigma$, by $\mathit{flat}(t) = \gamma_1 \gamma_2 \ldots \gamma_n$, is bijective. In particular, $\mathit{flat}(\alpha) = \varepsilon$. In the usual way we lift $\mathit{flat}$ to a bijection $\mathit{flat} : \mathcal{P}(T_\Sigma) \to \mathcal{P}((\Sigma - \{\alpha\})^*)$.

We define the set of *positions in a tree* by means of the mapping $\mathit{pos} : T_\Sigma(H) \to \mathcal{P}(\mathbb{N}^*)$ inductively as follows: (i) if $t \in \Sigma^{(0)} \cup H$, then $\mathit{pos}(t) = \{\varepsilon\}$, and (ii) if $t = \sigma(s_1, \ldots, s_k)$ where $\sigma \in \Sigma^{(k)}$, $k \geq 1$ and $s_1, \ldots, s_k \in T_\Sigma(H)$, then $\mathit{pos}(t) = \{\varepsilon\} \cup \{iv \mid 1 \leq i \leq k, v \in \mathit{pos}(s_i)\}$. For two positions $v, w \in \mathit{pos}(t)$, we say that $v$ *is a prefix of* $w$, denoted by $v \leq_{pre} w$, if there is a $u \in \mathbb{N}^*$ such that $vu = w$. Then $v^{-1}w$ denotes the string $u$.

Let $\alpha \in \Sigma^{(0)}$ and $t \in T_\Sigma$. Assume that $t$ contains $r \geq 0$ occurrences of $\alpha$. Moreover, let $u_1, \ldots, u_r \in T_\Sigma$. Define the tree $t[\alpha \leftarrow (u_1, \ldots, u_r)]$ as the tree $t' \in T_\Sigma$ which is obtained from $t$ by replacing the $i$-th occurrence of $\alpha$ (counted from the left to the right) by $u_i$. Then, $t$ is also called a *prefix* of $t'$.

Also, for a tree $t$ and one of its positions $w$, we define the *label of* $t$ *at* $w$, the *subtree of* $t$ *at* $w$, and the *replacement in* $t$ *at* $w$ as values of the mappings

$$\mathit{lab} : \{(t, w) \mid t \in T_\Sigma(H), w \in \mathit{pos}(t)\} \to \Sigma$$

and

$$\mathit{sub} : \{(t, w) \mid t \in T_\Sigma(H), w \in \mathit{pos}(t)\} \to T_\Sigma(H)$$

and

$$\mathit{repl} : \{(t, w) \mid t \in T_\Sigma(H), w \in \mathit{pos}(t)\} \times T_\Sigma(H) \to T_\Sigma(H),$$

resp., as follows: (i) if $t \in \Sigma^{(0)} \cup H$ (and thus $w = \varepsilon$), then $\mathit{lab}(t, \varepsilon) = \mathit{sub}(t, \varepsilon) = t$ and $\mathit{repl}(t, \varepsilon, t') = t'$, and (ii) if $t = \sigma(t_1, \ldots, t_k)$ for some $\sigma \in \Sigma^{(k)}$ with $k \geq 1$ and $t_1, \ldots, t_k \in T_\Sigma(H)$, then

- $\mathit{lab}(t, \varepsilon) = \sigma$, $\mathit{sub}(t, \varepsilon) = t$, and $\mathit{repl}(t, \varepsilon, t') = t'$,
- if $w = iv$ for some $1 \leq i \leq k$, then $\mathit{lab}(t, w) = \mathit{lab}(t_i, v)$, $\mathit{sub}(t, w) = \mathit{sub}(t_i, v)$, and $\mathit{repl}(t, w, t') = \sigma(t_1, ..., t_{i-1}, \mathit{repl}(t_i, v, t'), t_{i+1}, ..., t_k)$.

Let $V \subseteq \Sigma \cup H$ and $t \in T_\Sigma(H)$. Then $\mathit{pos}_V(t) = \{w \in \mathit{pos}(t) \mid \mathit{lab}(t, w) \in V\}$, comprising all positions of $t$ labelled with an element of $V$.

Clearly, for every $t \in T_\Sigma(H)$ and $w \in \mathit{pos}(t)$, $t = \mathit{repl}(t, w, \mathit{sub}(t, w))$. Instead of $\mathit{repl}(t, w, t')$ we also write $t[w \leftarrow t']$. If $w_1, \ldots, w_n \in \mathit{pos}(t)$ such that for every $w_i, w_j$ neither $w_i \leq_{pre} w_j$ nor $w_j \leq_{pre} w_i$, and $t_1, \ldots, t_n \in T_\Sigma(H)$, then $(\ldots((t[w_1 \leftarrow t_1])[w_2 \leftarrow t_2]) \ldots [w_n \leftarrow t_n])$ is abbreviated by $t[w_1 \leftarrow t_1, \ldots, w_n \leftarrow t_n]$. Also, instead of $\mathit{sub}(t, w)$ we write $t|_w$.

## 2.4 Tree languages and operations on tree languages

Every subset $L \subseteq T_\Sigma$ is called *(formal) tree language (over $\Sigma$)*. For two tree languages $L_1, L_2 \subseteq T_\Sigma$ and $\alpha \in \Sigma^{(0)}$, the *$\alpha$-concatenation of $L_1$ and $L_2$* is the tree language $L_1 \cdot_\alpha L_2 \subseteq T_\Sigma$ defined as follows (cf. page 66 of [TW68]; also cf. $\alpha$-product of $L_1$ and $L_2$ in Chapter II, Definition 4.5 of [GS84], and on page 16 of [GS97]):

$$L_1 \cdot_\alpha L_2 = \bigcup_{s \in L_1} s \cdot_\alpha L_2$$

where, for every $s \in T_\Sigma$ and $L \subseteq T_\Sigma$, if $s$ has $r \geq 0$ occurrences of $\alpha$, then $s \cdot_\alpha L = \{s[\alpha \leftarrow (u_1, \ldots, u_r)] \,|\, u_1, \ldots, u_r \in L\}$. Intuitively, $t \in s \cdot_\alpha L$ iff $t$ is obtained from $s$ by replacing all of the $\alpha$-leaves of $s$ by trees from $L$ (and keeping the rest of $s$ unchanged).

As special case of Corollary 2.4.2 of [ES77] (which states the associativity of $OI$-substitution of tree languages) we obtain the associativity of the $\alpha$-concatenation.

**Lemma 2.4** Let $L_1, L_2, L_3 \subseteq T_\Sigma$ be tree languages. Then $L_1 \cdot_\alpha (L_2 \cdot_\alpha L_3) = (L_1 \cdot_\alpha L_2) \cdot_\alpha L_3$.

In fact, the $\alpha$-concatenation is a straightforward generalization of the usual concatenation of $L_1 \cdot L_2$ of formal string languages $L_1$ and $L_2$.

**Observation 2.5** Let $\Sigma$ be an $\alpha$-string alphabet and $L_1, L_2 \subseteq T_\Sigma$. Then, $flat(L_1 \cdot_\alpha L_2) = flat(L_1) \cdot flat(L_2)$.

Let $L \subseteq T_\Sigma$ and $\alpha \in \Sigma^{(0)}$. We recall two classical definitions of $\alpha$-iterations of $L$. According to page 38 of [Eng75b], we define the *$\alpha$-E-iteration of* $L$ as the tree language $L_\alpha^{*,E} \subseteq T_\Sigma$ where $L_\alpha^{*,E} = \bigcup_{n \geq 0} L_\alpha^{n,E}$ with
(i) $L_\alpha^{0,E} = \{\alpha\}$ and
(ii) for every $n \geq 0$, $L_\alpha^{n+1,E} = L_\alpha^{n,E} \cdot_\alpha (L \cup \{\alpha\})$.

According to page 66 of [TW68] and page 17 of [GS97], we define the *$\alpha$-TW-iteration of* $L$ as the tree language $L_\alpha^{*,TW} \subseteq T_\Sigma$ where $L_\alpha^{*,TW} = \bigcup_{n \geq 0} L_\alpha^{n,TW}$ with
(i) $L_\alpha^{0,TW} = \{\alpha\}$ and
(ii) for every $n \geq 0$, $L_\alpha^{n+1,TW} = (L \cup \{\alpha\}) \cdot_\alpha L_\alpha^{n,TW}$.
Note that, for every $n \geq 0$, $L_\alpha^{n,E} \subseteq L_\alpha^{n+1,E}$ and $L_\alpha^{n,TW} \subseteq L_\alpha^{n+1,TW}$.

Obviously, there is a difference between the $\alpha$-E-iteration of $L$ and the $\alpha$-TW-iteration of $L$ in the definition of the induction step. However, as is well-known, due to the associativity of $\alpha$-concatenation, both definitions are equivalent.

**Observation 2.6** Let $L \subseteq T_\Sigma$ and $\alpha \in \Sigma^{(0)}$.
1) For every $n \geq 0$, $L_\alpha^{n,E} = L_\alpha^{n,TW}$.
2) $L_\alpha^{*,E} = L_\alpha^{*,TW}$.

**Proof.** Statement 1 can be proved just as Lemma 2.13 below, taking $A = \mathbb{B}$. The associativity of the $\alpha$-concatenation is essential in this proof. Statement 2 then follows immediately. □

Due to this observation, in the sequel we will drop the modifiers $E$ and $TW$ from $L_\alpha^{*,E}$ and $L_\alpha^{*,TW}$, respectively, and just write $L_\alpha^*$ and talk about the *$\alpha$-iteration of tree languages.* We note that both $\alpha$-iterations are natural generalizations of the usual Kleene-star of formal string languages.

**Observation 2.7** Let $\Sigma$ be an $\alpha$-string alphabet and $L \subseteq T_\Sigma$. Then $flat(L_\alpha^*) = flat(L)^*$.

## 2.5 Formal tree series and operations on formal tree series

Let $\Sigma$ be a ranked alphabet. A formal power series $\varphi \in A\langle\langle T_\Sigma \rangle\rangle$ is also called *formal tree series.* In the following, however, we will drop the word 'formal' from 'formal tree series' and just say tree series.

Let $k \geq 0$ and $\sigma \in \Sigma^{(k)}$. The *top-concatenation with* $\sigma$ is the operation $top_\sigma : A\langle\langle T_\Sigma \rangle\rangle^k \to A\langle\langle T_\Sigma \rangle\rangle$ such that, for every $\varphi_1, \ldots, \varphi_k \in A\langle\langle T_\Sigma \rangle\rangle$ and $t \in T_\Sigma$, if $t = \sigma(s_1, \ldots, s_k)$ with $s_1, \ldots, s_k \in T_\Sigma$, then $(top_\sigma(\varphi_1, \ldots, \varphi_k), t) = (\varphi_1, s_1) \odot \ldots \odot (\varphi_k, s_k)$ otherwise $(top_\sigma(\varphi_1, \ldots, \varphi_k), t) = 0$.

Let $\varphi, \psi \in A\langle\langle T_\Sigma \rangle\rangle$ and $\alpha \in \Sigma^{(0)}$. The *$\alpha$-concatenation of $\varphi$ and $\psi$* is the tree series $\varphi \cdot_\alpha \psi \in A\langle\langle T_\Sigma \rangle\rangle$ defined for every $t \in T_\Sigma$ by

$$(\varphi \cdot_\alpha \psi, t) = \sum_{\substack{s, u_1, \ldots, u_r \in T_\Sigma \\ t = s[\alpha \leftarrow (u_1, \ldots, u_r)]}} (\varphi, s) \odot (\psi, u_1) \odot \ldots \odot (\psi, u_r).$$

Note that, in the index set of the sum in the above observation we have $r \geq 0$. Observe that for every $a \in A$ and $\psi \in A\langle\langle T_\Sigma \rangle\rangle$ we have $a \cdot \psi = a\alpha \cdot_\alpha \psi$. We also note that $\varphi \cdot_\alpha \psi$ is well defined for arbitrary semirings (which need not be complete), because the involved sums are finite: for every $t, s \in T_\Sigma$ such that $t = s[\alpha \leftarrow (u_1, \ldots, u_r)]$, $s$ is a prefix of $t$, and $t$ has only finitely many prefixes.

Actually, the $\alpha$-concatenation of $\varphi$ and $\psi$ is a straightforward generalization of both, the $\alpha$-concatenation of tree languages and of the Cauchy product of formal power series. In the following observation we assume that, for an $\alpha$-string alphabet $\Sigma$, the bijection $flat$ has been extended in the obvious way to the mapping $flat : A\langle\langle T_\Sigma \rangle\rangle \to A\langle\langle (\Sigma - \{\alpha\})^* \rangle\rangle$ by defining for every $w \in (\Sigma - \{\alpha\})^*$, $(flat(\varphi), w) = (\varphi, flat^{-1}(w))$.

**Theorem 2.8** Let $\varphi, \psi \in A\langle\langle T_\Sigma \rangle\rangle$ and $\alpha \in \Sigma^{(0)}$.

1) If $A = \mathbb{B}$, then $supp(\varphi \cdot_\alpha \psi) = supp(\varphi) \cdot_\alpha supp(\psi)$.

2) If $\Sigma$ is an $\alpha$-string alphabet, then $flat(\varphi \cdot_\alpha \psi) = flat(\varphi) \cdot_C flat(\psi)$.

**Proof.** The proof of Statement 1 is an easy calculation:

$$\begin{array}{ll}
 & t \in supp(\varphi \cdot_\alpha \psi) \\
\text{iff} & (\varphi \cdot_\alpha \psi, t) = 1 \\
\text{iff} & \sum_{\substack{s, u_1, \ldots, u_r \in T_\Sigma \\ t = s[\alpha \leftarrow (u_1, \ldots, u_r)]}} (\varphi, s) \odot (\psi, u_1) \odot \ldots \odot (\psi, u_r) = 1 \\
\text{iff} & \text{there are } s, u_1, \ldots, u_r \in T_\Sigma \text{ such that } (\varphi, s) = (\psi, u_1) = \ldots = (\psi, u_r) = 1 \text{ and} \\
 & t = s[\alpha \leftarrow (u_1, \ldots, u_r)] \\
\text{iff} & \text{there are } s \in supp(\varphi),\ u_1, \ldots, u_r \in supp(\psi) \text{ and } t = s[\alpha \leftarrow (u_1, \ldots, u_r)] \\
\text{iff} & t \in supp(\varphi) \cdot_\alpha supp(\psi)
\end{array}$$

Let us assume that $\Sigma$ is an $\alpha$-string alphabet and prove Statement 2. Let $w \in (\Sigma - \{\alpha\})^*$.

$$\begin{array}{ll}
 & (flat(\varphi \cdot_\alpha \psi), w) \\
= & (\varphi \cdot_\alpha \psi, flat^{-1}(w)) \\
= & \sum_{\substack{s, u_1 \in T_\Sigma \\ flat^{-1}(w) = s[\alpha \leftarrow (u_1)]}} (\varphi, s) \odot (\psi, u_1) \\
 & \text{(note that } s \text{ contains exactly one } \alpha\text{, because } \Sigma \text{ is an } \alpha\text{-string alphabet)} \\
= & \sum_{\substack{u, v \in (\Sigma - \{\alpha\})^* \\ flat^{-1}(w) = flat^{-1}(u)[\alpha \leftarrow (flat^{-1}(v))]}} (\varphi, flat^{-1}(u)) \odot (\psi, flat^{-1}(v)) \\
= & \sum_{\substack{u, v \in (\Sigma - \{\alpha\})^* \\ w = uv}} (flat(\varphi), u) \odot (flat(\psi), v) \\
= & (flat(\varphi) \cdot_C flat(\psi), w)
\end{array}$$

□

As an immediate consequence we obtain:

**Corollary 2.9** 1) For every $\alpha \in \Sigma^{(0)}$, the mapping $supp : (\mathbb{B}\langle\langle T_\Sigma \rangle\rangle, \oplus, \cdot_\alpha, \tilde{0}, 1\alpha) \to (\mathcal{P}(T_\Sigma), \cup, \cdot_\alpha, \emptyset, \{\alpha\})$ is an isomorphism.

2) If $\Sigma$ is an $\alpha$-string alphabet, then $flat : (A\langle\langle T_\Sigma \rangle\rangle, \oplus, \cdot_\alpha, \tilde{0}, 1\alpha) \to (A\langle\langle (\Sigma - \{\alpha\})^* \rangle\rangle, \oplus, \cdot_C, \tilde{0}, 1\varepsilon)$ is an isomorphism.

Note that the structures in Corollary 2.9 do not satisfy all semiring axioms (see below). Hence, here the term 'isomorphism' has the usual meaning as in model theory or universal algebra, i.e., a bijection preserving all operations and constants.

**Lemma 2.10** Let $A$ be commutative. The $\alpha$-concatenation of tree series is associative and right-distributive, i.e., for every $\varphi, \psi, \rho \in A\langle\langle T_\Sigma \rangle\rangle$ and $\alpha \in \Sigma^{(0)}$, $\varphi \cdot_\alpha (\psi \cdot_\alpha \rho) = (\varphi \cdot_\alpha \psi) \cdot_\alpha \rho$ and $(\varphi \oplus \psi) \cdot_\alpha \rho = \varphi \cdot_\alpha \rho \oplus \psi \cdot_\alpha \rho$ .

**Proof.** Let $t \in T_\Sigma$. We calculate as follows:

$$\begin{aligned}
& (\varphi \cdot_\alpha (\psi \cdot_\alpha \rho), t) \\
= \; & \sum_{\substack{s,u_1,\ldots,u_r \in T_\Sigma \\ t = s[\alpha \leftarrow (u_1,\ldots,u_r)]}} (\varphi, s) \odot (\psi \cdot_\alpha \rho, u_1) \odot \ldots \odot (\psi \cdot_\alpha \rho, u_r) \\
= \; & \sum_{t = s[\alpha \leftarrow (u_1,\ldots,u_r)]} (\varphi, s) \odot \prod_{i=1}^{r} \sum_{u_i = v_i[\alpha \leftarrow (w_{i1},\ldots,w_{in_i})]} (\psi, v_i) \odot (\rho, w_{i1}) \odot \ldots \odot (\rho, w_{in_i}) \\
= \; & \sum_{\substack{t = s[\alpha \leftarrow (u_1,\ldots,u_r)] \\ u_i = v_i[\alpha \leftarrow (w_{i1},\ldots,w_{in_i})]}} (\varphi, s) \odot (\psi, v_1) \odot \ldots \odot (\psi, v_r) \odot \prod_{i=1}^{r} (\rho, w_{i1}) \odot \ldots \odot (\rho, w_{in_i}) \\
& \text{(by distributivity and commutativity)}
\end{aligned}$$

and

$$\begin{aligned}
& ((\varphi \cdot_\alpha \psi) \cdot \rho, t) \\
= \; & \sum_{\substack{v,w_1,\ldots,w_q \in T_\Sigma \\ t = v[\alpha \leftarrow (w_1 \ldots, w_q)]}} (\varphi \cdot_\alpha \psi, v) \odot (\rho, w_1) \odot \ldots \odot (\rho, w_q) \\
= \; & \sum_{t = v[\alpha \leftarrow (w_1 \ldots, w_q)]} \Big(\sum_{v = s[\alpha \leftarrow (v_1,\ldots,v_r)]} (\varphi, s) \odot (\psi, v_1) \odot \ldots \odot (\psi, v_r)\Big) \odot (\rho, w_1) \odot \ldots \odot (\rho, w_q) \\
= \; & \sum_{t = (s[\alpha \leftarrow (v_1,\ldots,v_r)])[\alpha \leftarrow (w_1,\ldots,w_q)]} (\varphi, s) \odot (\psi, v_1) \odot \ldots \odot (\psi, v_r) \odot (\rho, w_1) \odot \ldots \odot (\rho, w_q)
\end{aligned}$$

where the sums are taken over all possible decompositions of trees. Since there is a one-to-one correspondence between the two kinds of decomposing $t$, the above two sums are equal. Right-distributivity follows in a similar (easier) way. $\square$

Note that, as in the tree case, concatenation is not left-distributive, i.e., in general, $\varphi \cdot_\alpha (\psi \oplus \rho) \neq \varphi \cdot_\alpha \psi \oplus \varphi \cdot_\alpha \rho$. For instance, if $\varphi = 1\sigma(\alpha, \alpha)$, $\psi = 1\beta_1$, and $\rho = 1\beta_2$, then $(\varphi \cdot_\alpha (\psi \oplus \rho), \sigma(\beta_1, \beta_2)) = 1$ and $(\varphi \cdot_\alpha \psi \oplus \varphi \cdot_\alpha \rho, \sigma(\beta_1, \beta_2)) = 0$. However, if $\Sigma$ is an $\alpha$-string alphabet, then it follows immediately from Corollary 2.9 that concatenation is even left-distributive.

**Observation 2.11** Let $\Sigma$ be an $\alpha$-string alphabet and $\varphi, \psi, \rho \in A\langle\langle T_\Sigma \rangle\rangle$. Then $\varphi \cdot_\alpha (\psi \oplus \rho) = \varphi \cdot_\alpha \psi \oplus \varphi \cdot_\alpha \rho$.

As generalizations of the $\alpha$-E-iteration and the $\alpha$-TW-iteration of a tree language $L$, we now define also two different $\alpha$-iterations of tree series.

**Definition 2.12** Let $\varphi \in A\langle\langle T_\Sigma \rangle\rangle$ and $\alpha \in \Sigma^{(0)}$. The *$n$-th $\alpha$-E-iteration of* $\varphi$ is the tree series $\varphi_\alpha^{n,E} \in A\langle\langle T_\Sigma \rangle\rangle$ defined inductively as follows:

(i) $\varphi_\alpha^{0,E} = 1\,\alpha$ and

(ii) for every $n \geq 0$, $\varphi_\alpha^{n+1,E} = \varphi_\alpha^{n,E} \cdot_\alpha (\varphi \oplus 1\,\alpha)$

and the *$n$-th $\alpha$-TW-iteration of* $\varphi$ is the tree series $\varphi_\alpha^{n,TW} \in A\langle\langle T_\Sigma \rangle\rangle$ defined inductively as follows:

(i) $\varphi_\alpha^{0,TW} = 1\,\alpha$ and

(ii) for every $n \geq 0$, $\varphi_\alpha^{n+1,TW} = (\varphi \oplus 1\,\alpha) \cdot_\alpha \varphi_\alpha^{n,TW}$. $\square$

Notice the similarity of these definitions to the corresponding ones for tree languages, but the difference to the definition of the Kleene-powers of series over words. It is this difference which will later on make it necessary for us to assume that $A$ is idempotent. First we show:

**Lemma 2.13** Let $A$ be commutative. For every $\varphi \in A\langle\langle T_\Sigma \rangle\rangle$ and $n \geq 0$, we have $\varphi_\alpha^{n,E} = \varphi_\alpha^{n,TW}$.

**Proof.** By associativity of the $\alpha$-concatenation, for every $n, m \geq 0$, we have

$$(*) \quad \varphi_\alpha^{n+1,E} \cdot_\alpha \varphi_\alpha^{m,TW} = (\varphi_\alpha^{n,E} \cdot_\alpha (\varphi \oplus 1\,\alpha)) \cdot_\alpha \varphi_\alpha^{m,TW} = \varphi_\alpha^{n,E} \cdot_\alpha ((\varphi \oplus 1\,\alpha) \cdot_\alpha \varphi_\alpha^{m,TW}) = \varphi_\alpha^{n,E} \cdot_\alpha \varphi_\alpha^{m+1,TW}.$$

Now applying (*) $n$ times, for every $n \geq 1$, we obtain $\varphi_\alpha^{n,E} = \varphi_\alpha^{n,E} \cdot_\alpha 1\,\alpha = \varphi_\alpha^{n,E} \cdot_\alpha \varphi_\alpha^{0,TW} =^{(*)} \varphi_\alpha^{0,E} \cdot_\alpha \varphi_\alpha^{n,TW} = 1\,\alpha \cdot_\alpha \varphi_\alpha^{n,TW} = \varphi_\alpha^{n,TW}$. □

Since we only want to deal with one iteration of tree series, we assume in the sequel that $A$ is commutative and we abbreviate $\varphi_\alpha^{n,TW}$ and $\varphi_\alpha^{n,E}$ by $\varphi_\alpha^n$. We define the star of a tree series $\varphi$ at $\alpha$ to be the sum of $\alpha$-iterations of $\varphi$. More precisely, given a tree $t \in T_\Sigma$, then we sum up all the coefficients $(\varphi_\alpha^n, t)$ for $0 \leq n \leq size_{\neg\alpha}(t)$, where $size_{\neg\alpha}(t) = size(t) - card(pos_{\{\alpha\}}(t)) = card(pos_{\Sigma-\{\alpha\}}(t))$, i.e., the number of non-alpha labeled positions of $t$.

**Definition 2.14** Let $A$ be commutative, $\varphi \in A\langle\langle T_\Sigma\rangle\rangle$, and $t \in T_\Sigma$. Then we define the *Kleene-star* $\varphi_\alpha^* \in A\langle\langle T_\Sigma\rangle\rangle$ of $\varphi$ (at $\alpha$) as follows:

$$(\varphi_\alpha^*, t) = \sum_{n=0}^{size_{\neg\alpha}(t)} (\varphi_\alpha^n, t).$$ □

**Example 2.15** Let $\Sigma = \{\sigma^{(2)}, \alpha^{(0)}, \beta^{(0)}\}$, $A = \mathbb{B}$, and $\varphi = 1\,\sigma(\alpha,\alpha) \vee 1\,\beta$. Then, since $size_{\neg\alpha}(\sigma(\alpha,\beta)) = 2$ we can compute as follows:

$(\varphi_\alpha^*, \sigma(\alpha,\beta)) = \sum_{n=0}^{size_{\neg\alpha}(\sigma(\alpha,\beta))} (\varphi_\alpha^n, \sigma(\alpha,\beta)) = (\varphi_\alpha^0, \sigma(\alpha,\beta)) \vee (\varphi_\alpha^1, \sigma(\alpha,\beta)) \vee (\varphi_\alpha^2, \sigma(\alpha,\beta)) = (1\,\alpha, \sigma(\alpha,\beta)) \vee (1\,\sigma(\alpha,\alpha) \vee 1\,\beta \vee 1\,\alpha, \sigma(\alpha,\beta)) \vee (\ldots \vee 1\,(\alpha[\alpha \leftarrow (\sigma(\alpha,\alpha)])[\alpha \leftarrow (\alpha,\beta)] \vee \ldots, \sigma[\alpha,\beta)) = 0 \vee 0 \vee 1 = 1$. □

Let $\alpha \in \Sigma^{(0)}$. A tree series $\varphi \in A\langle\langle T_\Sigma\rangle\rangle$ is *$\alpha$-unitary* if $(\varphi, \alpha) = 1$. The class of all $\alpha$-unitary tree series over $\Sigma$ and $A$ is denoted by $A_\alpha\langle\langle T_\Sigma\rangle\rangle$.

We will show (see Lemma 2.17) that, if $A$ is also idempotent (as it will be in all our subsequent considerations of the star) and $\varphi$ is $\alpha$-unitary, then the definition of the Kleene-star can be simplified to $(\varphi_\alpha^*, t) = \sum_{n=0}^{size(t)} (\varphi_\alpha^n, t) = (\varphi_\alpha^{size(t)}, t)$.

**Lemma 2.16** Let $A$ be commutative and idempotent, $\alpha \in \Sigma^{(0)}$, $\varphi \in A_\alpha\langle\langle T_\Sigma\rangle\rangle$, and $n \geq 0$. Then the following statements are true.

1. $\varphi_\alpha^{n+1} = \varphi_\alpha^n \cdot_\alpha \varphi = \varphi \cdot_\alpha \varphi_\alpha^n = \varphi_\alpha^{n+1} \oplus \varphi_\alpha^n = \sum_{i=0}^{n+1} \varphi_\alpha^i$.
2. $\varphi_\alpha^n$ is $\alpha$-unitary.

**Proof.** 1) Due to the fact that $\varphi$ is $\alpha$-unitary and $A$ is idempotent, it follows that $\varphi \oplus 1\,\alpha = \varphi$. Hence, by Definition 2.12 and Lemma 2.13, $\varphi_\alpha^{n+1} = \varphi_\alpha^{n+1,E} = \varphi_\alpha^{n,E} \cdot_\alpha \varphi = \varphi_\alpha^n \cdot_\alpha \varphi$ and $\varphi_\alpha^{n+1} = \varphi_\alpha^{n+1,TW} = \varphi \cdot_\alpha \varphi_\alpha^{n,TW} = \varphi \cdot_\alpha \varphi_\alpha^n$ and $\varphi_\alpha^{n+1} = (\varphi \oplus 1\alpha) \cdot_\alpha \varphi_\alpha^n = \varphi_\alpha^{n+1} \oplus \varphi_\alpha^n$. And, due to idempotency of $A$, $\varphi_\alpha^{n+1} \oplus \varphi_\alpha^n = \sum_{i=0}^{n+1} \varphi_\alpha^i$.

2) For $n = 0$ this is trivial. Assuming that $\varphi_\alpha^n$ is $\alpha$-unitary, we can calculate as follows: $(\varphi_\alpha^{n+1}, \alpha) = (\varphi \cdot_\alpha \varphi_\alpha^n, \alpha) = (\varphi, \alpha) \odot (\varphi_\alpha^n, \alpha) = 1 \odot 1 = 1$. □

**Lemma 2.17** Let $A$ be commutative and idempotent, $\varphi \in A_\alpha\langle\langle T_\Sigma\rangle\rangle$, $t \in T_\Sigma$.

1. If $n \geq size_{\neg\alpha}(t)$, then $(\varphi_\alpha^{n+1}, t) = (\varphi_\alpha^n, t)$.
2. If $l, r \geq size_{\neg\alpha}(t)$, then $\sum_{n=0}^{l} (\varphi_\alpha^n, t) = \sum_{n=0}^{r} (\varphi_\alpha^n, t)$.
3. $(\varphi_\alpha^*, t) = (\varphi_\alpha^{size_{\neg\alpha}(t)}, t) = (\varphi_\alpha^{size(t)}, t)$.

**Proof.** We show Statement 1 by induction on the size of $t$. If $t = \alpha$, then the result is immediate by Lemma 2.16(2). If $t = \beta \in \Sigma^{(0)} - \{\alpha\}$, we claim that $(\varphi_\alpha^n, \beta) = (\varphi, \beta)$ for every $n \geq 1$. By induction on $n$ and Lemma 2.16 we obtain: $(\varphi_\alpha^{n+1}, \beta) = (\varphi_\alpha^n \cdot_\alpha \varphi, \beta) = (\varphi_\alpha^n, \beta) \oplus (\varphi_\alpha^n, \alpha) \odot (\varphi, \beta) =$ (induction hyp.) $(\varphi, \beta) \oplus (\varphi, \beta) =$ (idempotency) $(\varphi, \beta)$.

Now assume that $size(t) \geq 2$ and $n \geq size_{\neg\alpha}(t)$. We calculate

$$
\begin{array}{rl}
& (\varphi_\alpha^{n+1}, t) \\
= & (\varphi_\alpha^n \cdot_\alpha \varphi, t) \\
= & \sum_{\substack{s,u_1,\ldots,u_r \in T_\Sigma \\ t = s[\alpha \leftarrow (u_1,\ldots,u_r)]}} (\varphi_\alpha^n, s) \odot (\varphi, u_1) \odot \ldots \odot (\varphi, u_r) \\
= & (\varphi_\alpha^n, t) \oplus \sum_{\substack{t = s[\alpha \leftarrow (u_1,\ldots,u_r)] \\ s \neq t}} (\varphi_\alpha^n, s) \odot (\varphi, u_1) \odot \ldots \odot (\varphi, u_r)
\end{array}
$$

If $t = s[\alpha \leftarrow (u_1, \ldots, u_r)]$ and $s \neq t$, then $size_{\neg\alpha}(s) < size_{\neg\alpha}(t)$, and hence, $size_{\neg\alpha}(s) \leq n-1$. By induction hypothesis, $(\varphi_\alpha^n, s) = (\varphi_\alpha^{n-1}, s)$. Since $\varphi_\alpha^n = \varphi_\alpha^n \oplus \varphi_\alpha^{n-1}$, the above expression equals

$$
\begin{array}{rl}
& (\varphi_\alpha^n, t) \oplus (\varphi_\alpha^{n-1}, t) \oplus \sum_{\substack{t = s[\alpha \leftarrow (u_1,\ldots,u_r)] \\ s \neq t}} (\varphi_\alpha^{n-1}, s) \odot (\varphi, u_1) \odot \ldots \odot (\varphi, u_r) \\
= & (\varphi_\alpha^n, t) \oplus (\varphi_\alpha^{n-1} \cdot_\alpha \varphi, t) \\
= & (\varphi_\alpha^n, t)
\end{array}
$$

Statement 2 follows directly from Statement 1 and the idempotency of $A$. Statement 3 for $t = \alpha$ is trivial. Let $t \neq \alpha$, then $t$ contains at least one node which is not labeled by $\alpha$, hence $size_{\neg\alpha}(t) \geq 1$. Then, by Lemma 2.16(1) and Statement 1 of the present lemma, we have $(\varphi_\alpha^*, t) = \sum_{n=0}^{size_{\neg\alpha}(t)} (\varphi_\alpha^n, t) = (\varphi_\alpha^{size_{\neg\alpha}(t)}, t) = (\varphi_\alpha^{size(t)}, t)$. $\square$

**Lemma 2.18** Let $A$ be commutative and idempotent, $\alpha \in \Sigma^{(0)}$, $\varphi \in A_\alpha\langle\langle T_\Sigma \rangle\rangle$, and $n \geq 0$. Then $\varphi_\alpha^* \cdot_\alpha \varphi = \varphi \cdot_\alpha \varphi_\alpha^* = \varphi_\alpha^*$.

**Proof.** Let $t \in T_\Sigma$. For $n \geq size(t)$ we calculate as follows: $(\varphi_\alpha^*, t) = (\varphi_\alpha^{n+1}, t) = (\varphi_\alpha^n \cdot_\alpha \varphi, t) = (\varphi_\alpha^* \cdot_\alpha \varphi, t)$. In a similar way we can prove the statement $\varphi \cdot_\alpha \varphi_\alpha^* = \varphi_\alpha^*$. $\square$

Assuming commutativity and idempotency, the iteration of tree series is a natural generalization of both, the iteration of tree languages and the Kleene-star of formal power series. Here we have to deal with the different definitions of the $(n+1)$st Kleene power and the $(n+1)$st iteration which is also reflected by the following auxiliary statement.

**Lemma 2.19** Let $A$ be commutative and idempotent, $\Sigma$ be an $\alpha$-string alphabet, and $\varphi \in A_\alpha\langle\langle T_\Sigma \rangle\rangle$. Then, for every $n \geq 1$, $flat(\varphi_\alpha^n) = \sum_{i=0}^n flat(\varphi)[\varepsilon/0]^{i,K}$.

**Proof.** We prove the claim by induction on $n$. If $n = 0$, we have $flat(\varphi_\alpha^0) = flat(1\alpha) = 1\varepsilon = flat(\varphi)[\varepsilon/0]^{0,K}$. For the induction step we derive:

$$
\begin{array}{rl}
& flat(\varphi_\alpha^{n+1}) \\
= & flat(\varphi_\alpha^n \cdot_\alpha \varphi) \\
& \text{(by Lemma 2.16)} \\
= & flat(\varphi_\alpha^n) \cdot_C flat(\varphi) \\
& \text{(by Statement 2 of Theorem 2.8)} \\
= & flat(\varphi_\alpha^n) \cdot_C flat(\varphi)[\varepsilon/0] \oplus flat(\varphi_\alpha^n) \\
& \text{(by Lemma 2.2; note that } (flat(\varphi), \varepsilon) = 1) \\
= & (\sum_{i=0}^n flat(\varphi)[\varepsilon/0]^{i,K}) \cdot_C flat(\varphi)[\varepsilon/0] \oplus flat(\varphi_\alpha^n) \\
& \text{(by induction hypothesis)} \\
= & \sum_{i=0}^n (flat(\varphi)[\varepsilon/0]^{i,K} \cdot_C flat(\varphi)[\varepsilon/0]) \oplus flat(\varphi_\alpha^n) \\
= & \sum_{i=0}^n flat(\varphi)[\varepsilon/0]^{i+1,K} \oplus \sum_{i=0}^n flat(\varphi)[\varepsilon/0]^{i,K} \\
& \text{(by induction hypothesis)} \\
= & \sum_{i=0}^{n+1} flat(\varphi)[\varepsilon/0]^{i,K} \\
& \text{(by idempotency of } A)
\end{array}
$$

$\square$

Now we can show:

**Theorem 2.20** Let $A$ be commutative and idempotent, $\alpha \in \Sigma^{(0)}$, and $\varphi \in A\langle\langle T_\Sigma \rangle\rangle$.

1. If $A = \mathbb{B}$, then $supp(\varphi^{*\alpha}) = (supp(\varphi))^{*\alpha}$.
2. If $\Sigma$ is an $\alpha$-string alphabet and $\varphi$ is $\alpha$-unitary, then $flat(\varphi^*_\alpha) = flat(\varphi)[\varepsilon/0]^{*,K}$.

**Proof.** Statement 1 is straightforward using Corollary 2.9. Let us prove Statement 2. Let $w \in (\Sigma - \{\alpha\})^*$. Then, $size_{\neg\alpha}(flat^{-1}(w)) = length(w)$. Choose $n \geq length(w)$. By Lemma 2.17(1) and Lemma 2.19 we have:

$$\begin{aligned}
& (flat(\varphi^*_\alpha), w) \\
= \; & (\varphi^*_\alpha, flat^{-1}(w)) \\
= \; & (\varphi^n_\alpha, flat^{-1}(w)) \\
= \; & (flat(\varphi^n_\alpha), w) \\
= \; & (\textstyle\sum_{i=0}^n flat(\varphi)[\varepsilon/0]^{i,K}, w) \\
= \; & (flat(\varphi)[\varepsilon/0]^{*,K}, w)
\end{aligned}$$

□

For later use in Lemma 5.9 dealing with the iteration we show how to compute coefficients of the form $(\varphi^n_\alpha, t)$ explicitly. We prepare this by defining the set of $n$-decompositions of a tree $t$. Roughly speaking, $t$ is decomposed into $n$ layers where every layer consists of trees over $\Sigma$. Note that it is possible that a layer contains $\alpha$'s, and even that a layer only consists of $\alpha$'s.

**Definition 2.21** Let $t \in T_\Sigma$, $\alpha \in \Sigma^{(0)}$, and $n \geq 1$. The *set of $n$-decompositions of $t$* is the set

$$\begin{aligned}
M_{\alpha,n}(t) = \{(s, (u^i_j)_{\substack{n-1 \geq i \geq 1 \\ 1 \leq j \leq r_i}}) \mid s, u^i_j \in T_\Sigma, \text{ if } s_n = s \text{ and for every } n-1 \geq i \geq 1 : \\
s_{i+1}[\alpha \leftarrow (u^i_1, \ldots, u^i_{r_i})] = s_i, \text{ then } s_1 = t\}.
\end{aligned}$$

The *set of all decompositions of $t$* is the set $M_\alpha(t) = \bigcup_{n \geq 1} M_{\alpha,n}(t)$. □

Thus, a tree $t$ is decomposed in the following way:

$$\underbrace{\underbrace{\left( \cdots \underbrace{\left( \underbrace{s}_{s_n} [\alpha \leftarrow (u^{n-1}_1, \ldots, u^{n-1}_{r_{n-1}})] \right)}_{s_{n-1}} \cdots [\alpha \leftarrow (u^2_1, \ldots, u^2_{r_2})] \right)}_{s_2} [\alpha \leftarrow (u^1_1, \ldots, u^1_{r_1})]}_{s_1 = t}$$

and formally, an $n$-decomposition of $t$ has the form

$$(s, \underbrace{(u^{n-1}_1, \ldots, u^{n-1}_{r_{n-1}})}_{\text{(n-1)-th layer}}, \ldots, \underbrace{(u^2_1, \ldots, u^2_{r_2})}_{\text{2nd layer}}, \underbrace{(u^1_1, \ldots, u^1_{r_1})}_{\text{1st layer}})$$

where the 1st layer is close to the leaves of $t$. Thus, an $n$-decomposition of $t$ consists of a single tree $s$ and $n-1$ layers. Note that $M_{\alpha,n}(t)$ is a finite set.

**Example 2.22** Let $\Sigma = \{\sigma^{(2)}, \gamma^{(1)}, \alpha^{(0)}, \beta^{(0)}\}$ and $t = \sigma(\alpha, \gamma\gamma\beta)$. Then

$$(\underbrace{\sigma(\alpha,\alpha)}_{s}, (\underbrace{\alpha}_{u^3_1}, \underbrace{\gamma\alpha}_{u^3_2}), (\underbrace{\alpha}_{u^2_1}, \underbrace{\gamma\alpha}_{u^2_2}), (\underbrace{\alpha}_{u^1_1}, \underbrace{\beta}_{u^1_2})) \in M_{\alpha,4}(t).$$

Also,

$$\begin{aligned}
M_{\alpha,2}(\sigma(\alpha,\beta)) = \; & \{(\alpha, (\sigma(\alpha,\beta))),\ (\sigma(\alpha,\alpha), (\alpha,\beta)),\ (\sigma(\alpha,\beta), (\alpha))\} \\
M_{\alpha,3}(\sigma(\alpha,\beta)) = \; & \{(\alpha, (\alpha), (\sigma(\alpha,\beta))),\ (\alpha, (\sigma(\alpha,\alpha)), (\alpha,\beta)),\ (\alpha, (\sigma(\alpha,\beta)), (\alpha)), \\
& (\sigma(\alpha,\alpha), (\alpha,\alpha), (\alpha,\beta)),\ (\sigma(\alpha,\alpha), (\alpha,\beta), (\alpha)),\ (\sigma(\alpha,\beta), (\alpha)(\alpha))\}
\end{aligned}$$

**Definition 2.23** Let $t \in T_\Sigma$, $n \geq 0$, and $x = (s, (u^i_j)_{\substack{n-1 \geq i \geq 1 \\ 1 \leq i \leq r_i}}) \in M_{\alpha,n}(t)$. Moreover, let $\varphi \in A_\alpha\langle\langle T_\Sigma\rangle\rangle$. The *cost of $x$ with respect to $\varphi$* is the element

$$c_\varphi(x) = (\varphi, s) \odot \prod_{i=n-1}^{1} \prod_{j=1}^{r_i} (\varphi, u^i_j)$$

of $A$. □

Thus if $x = (s, (u_1^{n-1}, \ldots, u^{n-1}_{r_{n-1}}), \ldots, (u^2_1, \ldots, u^2_{r_2}), (u^1_1, \ldots, u^1_{r_1}))$, then

$$c_\varphi(x) = (\varphi, s) \odot \underbrace{(\varphi, u_1^{n-1}) \odot \ldots (\varphi, u^{n-1}_{r_{n-1}})}_{(n-1)-th\, layer} \odot \ldots \odot \underbrace{(\varphi, u^2_1) \odot \ldots \odot (\varphi, u^2_{r_2})}_{2nd\, layer} \odot \underbrace{(\varphi, u^1_1) \odot \ldots \odot (\varphi, u^1_{r_1})}_{1st\, layer}.$$

The following result gives the explicit description of the $n$-th iteration of a series $\varphi$ in terms of $n$-decompositions of trees.

**Lemma 2.24** Let $A$ be commutative and idempotent and $\varphi \in A_\alpha\langle\langle T_\Sigma\rangle\rangle$, $t \in T_\Sigma$, and $n \geq 1$. Then $(\varphi^n_\alpha, t) = \sum_{x \in M_{\alpha,n}(t)} c_\varphi(x)$.

**Proof.** We prove by induction on $n$. First, $(\varphi^1_\alpha, t) = (\varphi, t) = c_\varphi((t)) = \sum_{x \in M_{\alpha,1}(t)} c_\varphi(x)$. For the induction step we calculate:

$$\begin{aligned}
& (\varphi^{n+1}_\alpha, t) \\
= \; & (\varphi^n_\alpha \cdot_\alpha \varphi, t) \\
= \; & \sum_{\substack{s,u_1,\ldots,u_r \in T_\Sigma \\ t = s[\alpha \leftarrow (u_1,\ldots,u_r)]}} (\varphi^n_\alpha, s) \odot (\varphi, u_1) \odot \ldots \odot (\varphi, u_r) \\
= \; & \sum_{\substack{s,u_1,\ldots,u_r \in T_\Sigma \\ t = s[\alpha \leftarrow (u_1,\ldots,u_r)]}} (\textstyle\sum_{x \in M_{\alpha,n}(s)} c_\varphi(x)) \odot (\varphi, u_1) \odot \ldots \odot (\varphi, u_r) \\
& \text{(by induction hypothesis)} \\
= \; & \sum_{\substack{s,u_1,\ldots,u_r \in T_\Sigma \\ t = s[\alpha \leftarrow (u_1,\ldots,u_r)] \\ x \in M_{\alpha,n}(s)}} c_\varphi(x) \odot (\varphi, u_1) \odot \ldots \odot (\varphi, u_r) \\
= \; & \textstyle\sum_{y \in M_{\alpha,n+1}(t)} c_\varphi(y)
\end{aligned}$$

□

## 2.6 Rational Tree Series

Here we recall the definition of rational tree (series) expressions from [GS97]. However, rather than using an additional set $Z$ of symbols for substitution (as it was done in [GS97]) we use the symbols of $\Sigma^{(0)}$ for this purpose (cf. [Eng75b]). As usual, by means of rational tree series expressions, the class $A^{rat}\langle\langle T_\Sigma\rangle\rangle$ of rational tree series over $\Sigma$ and $A$ is defined as follows.

**Definition 2.25** The set of *rational tree series expressions over $\Sigma$ and $A$*, denoted by $Rat(\Sigma, A)$, and the *tree series* $[\![\eta]\!] \in A\langle\langle T_\Sigma\rangle\rangle$ denoted by such an expression $\eta$ are defined inductively by the following six conditions:

1. $\Sigma^{(0)} \subseteq Rat(\Sigma, A)$ and, for every $\alpha \in \Sigma^{(0)}$, $[\![\alpha]\!] = \alpha$.
2. For every $k \geq 1$, $\sigma \in \Sigma^{(k)}$, and $\eta_1, \ldots, \eta_k \in Rat(\Sigma, A)$, the expression $\sigma(\eta_1, \ldots, \eta_k) \in Rat(\Sigma, A)$ and $[\![\sigma(\eta_1, \ldots, \eta_k)]\!] = top_\sigma([\![\eta_1]\!], \ldots, [\![\eta_k]\!])$.
3. For every $\eta \in Rat(\Sigma, A)$ and $a \in A$, the expression $(a\eta) \in Rat(\Sigma, A)$ and $[\![(a\eta)]\!] = a \odot [\![\eta]\!]$.
4. For every $\eta_1, \eta_2 \in Rat(\Sigma, A)$, the expression $(\eta_1 + \eta_2) \in Rat(\Sigma, A)$ and $[\![(\eta_1 + \eta_2)]\!] = [\![\eta_1]\!] \oplus [\![\eta_2]\!]$.
5. For every $\eta_1, \eta_2 \in Rat(\Sigma, A)$ and $\alpha \in \Sigma^{(0)}$, the expression $(\eta_1 \cdot_\alpha \eta_2) \in Rat(\Sigma, A)$ and $[\![(\eta_1 \cdot_\alpha \eta_2)]\!] = [\![\eta_1]\!] \cdot_\alpha [\![\eta_2]\!]$.

6. For every $\eta \in Rat(\Sigma, A)$ and $\alpha \in \Sigma^{(0)}$ such that $[\![\eta]\!]$ is $\alpha$-unitary, the expression $(\eta_\alpha^*) \in Rat(\Sigma, A)$ and $[\![(\eta_\alpha^*)]\!] = [\![\eta]\!]_\alpha^*$. $\square$

**Definition 2.26**

1. A tree series $\varphi \in A\langle\!\langle T_\Sigma \rangle\!\rangle$ is called a *rational tree series over* $\Sigma$ *and* $A$ if there is an $\eta \in Rat(\Sigma, A)$ such that $\varphi = [\![\eta]\!]$.
2. The *class of rational tree series over* $\Sigma$ *and* $A$ is denoted by $A^{rat}\langle\!\langle T_\Sigma \rangle\!\rangle$. $\square$

We say that a class $\mathcal{C} \subseteq A\langle\!\langle T_\Sigma \rangle\!\rangle$ is *closed under the rational operations* if it is closed under top-concatenation $top_\sigma$ for every $\sigma \in \Sigma$, multiplication with coefficients in $A$, sum in $A\langle\!\langle T_\Sigma \rangle\!\rangle$, concatenation at $\alpha$ and iteration at $\alpha$ for every $\alpha \in \Sigma^{(0)}$. The following observation follows immediately from the Definition 2.25.

**Observation 2.27** $A^{rat}\langle\!\langle T_\Sigma \rangle\!\rangle$ is the smallest subclass of $A\langle\!\langle T_\Sigma \rangle\!\rangle$ which contains the class $A\langle T_\Sigma \rangle$ and is closed under the rational operations.

# 3 Recognizable Tree Series

In this section we define the notion of bottom-up weighted tree automata (also cf. [Sei94, Kui97a]) by means of which we obtain the class $A^{rec}\langle\!\langle T_\Sigma \rangle\!\rangle$ of recognizable tree series over $\Sigma$ and $A$. From now on and for the rest of this paper, $A$ *stands for a commutative semiring.*

**Definition 3.1** Let $A$ be a commutative semiring. A *bottom-up (finite state) weighted tree automaton over* $A$ is a quadruple $M = (Q, \Sigma, \delta, F)$ where $Q$ is a finite set (of states), $\Sigma$ is a ranked alphabet with $\Sigma \cap Q = \emptyset$, $F \subseteq Q$ is the set (of final states), $\delta = \{\delta_\sigma\}_{k \geq 0, \sigma \in \Sigma^{(k)}}$ is a family of state behaviours (or: rules, transitions) with costs such that, for every $k \geq 0$ and $\sigma \in \Sigma^{(k)}$, $\delta_\sigma : Q^k \times Q \to A$ is a mapping. $\square$

Usually, if $\delta_\sigma((q_1, \ldots, q_k), q) = a$, then we say that $M$ contains the rule $\sigma(q_1, \ldots, q_k) \to q$ with cost $a$. Note that, if $\sigma \in \Sigma^{(k)}$, then for every $q, q_1, \ldots, q_k \in Q$, $M$ contains a rule $\sigma(q_1, \ldots, q_k) \to q$ with some cost $a \in A$. The reason why we defined tree automata in a somehow unusual way is the fact that we do not want to distinguish between the following two cases: for some $\sigma \in \Sigma^{(k)}$ and $q_1, \ldots, q_k, q \in Q$, (1) there is no rules $\sigma(q_1, \ldots, q_k) \to q$ in $M$ and (2) there is a rule $\sigma(q_1, \ldots, q_k) \to q$ in $M$ which has cost 0. Clearly, whenever one of these two cases is used in a derivation, then the global cost will be 0. Note that it is not a viable solution to keep the usual rule based definition and just to require that the cost of a rule is $\neq 0$. In fact, if such an automaton is the result of some construction, then it might happen that the cost of a constructed rule is the product $a \odot b$ for some $a, b \in A$ and, due to the presence of zero-divisors in $A$, possibly $a \odot b = 0$.

In the rest of this section let $M = (Q, \Sigma, \delta, F)$ be a bottom-up weighted tree automaton over $A$. As usual, we will identify $Q^k \times Q$ with $Q^{k+1}$. Next we define the concept of runs of $M$, their costs, and the tree series accepted by $M$. Since, as noted above, $M$ contains all possible rules as transitions (possibly with costs 0) a run will just be any mapping from the set of positions of the given input tree $t$ to $Q$.

**Definition 3.2** 1. Let $P \subseteq Q$ and $t \in T_\Sigma(Q)$. A *run of* $M$ *on* $t$ *using* $P$ is a mapping $r : pos(t) \to Q$ such that

- $r(w) \in P$ for every $w \in pos(t)$-$(pos_Q(t) \cup \{\varepsilon\})$ and
- $r(w) = lab(t, w)$ for every $w \in pos_Q(t)$.

Then $r$ *reaches* the state $q \in Q$, if $r(\varepsilon) = q$. The *set of all runs of* $M$ *on* $t$ *using* $P$ *reaching* $q$ is denoted by $R_M^P(t, q)$. We abbreviate $R_M^Q(t, q)$ by $R_M(t, q)$ and define $R_M(t) = \bigcup_{q \in Q} R_M(t, q)$.

2. Let $w \in pos(t)$. The *cost* $c(t, r, w)$ *of* $w$ *in* $t$ *under* $r$ is defined by

$$c(t, r, w) = \begin{cases} \delta_\sigma(r(w1), \ldots, r(wk), r(w)) & \text{if } lab(t, w) = \sigma \in \Sigma^{(k)}, k \geq 0 \\ 1 & \text{if } lab(t, w) \in Q \end{cases}$$

and the *cost of r for t* is defined by

$$c(t,r) = \prod_{w \in pos(t)} c(t,r,w).$$

3. The *tree series accepted by M*, denoted by $S_M \in A\langle\langle T_\Sigma \rangle\rangle$, is the tree series defined, for every $t \in T_\Sigma$, by
$$(S_M, t) = \sum_{q \in F} \sum_{r \in R_M(t,q)} c(t,r).$$

4. Let $M$ and $M'$ be bottom-up weighted tree automata. Then $M$ and $M'$ are *equivalent* if $S_M = S_{M'}$.

5. A tree series $\varphi \in A\langle\langle T_\Sigma \rangle\rangle$ is called *recognizable tree series over* $\Sigma$ *and* $A$ if there is a bottom-up weighted tree automaton $M$ over $A$ such that $\varphi = S_M$.

6. The *class of all recognizable tree series over* $\Sigma$ *and* $A$ is denoted by $A^{rec}\langle\langle T_\Sigma \rangle\rangle$. □

We note that we could also define $c(t,r)$ if $A$ is not commutative, by deciding on a particular order in which the product is evaluated, i.e., by ordering the set $pos(t)$.

Later we will often have the task to decompose a run $d$ of a bottom-up tree automaton at those nodes for which a particular property $U$ holds. The idea of decomposing a tree $t$ and a run $r$ of $M$ on $t$ is to cut of the prefix $t'$ of $t$ which ends up in nodes of $t$ having the node property $U$. Then $r$ induces a run $r'$ on $t'$.

**Definition 3.3** Let $t \in T_\Sigma(Q)$. A *node property of* $t$ is a mapping $U : R_M(t) \to \mathcal{P}(pos(t))$.

**Definition 3.4** Let $t \in T_\Sigma(Q)$, $r : pos(t) \to Q$ be a run of $M$ on $t$, and $U$ be a node property of $t$.

Define the *U-decomposition of t and r* by

$$dec_U(t,r) = (t', r', (w_1, t_1, r_1), \ldots, (w_m, t_m, r_m))$$

such that the following conditions hold:

1. $\{w_1, \ldots, w_m\} = \{w \in pos(t) \,|\, w \in U(r), w \neq \varepsilon$ and for every $v \in pos(t)$ such that $v \leq_{pre} w$ and $v \neq \varepsilon$ and $v \neq w : v \notin U(r)\}$, i.e., the $w_i$'s are the topmost positions of $t$ (disregarding the root) for which the node property $U(r)$ holds,

2. $t' = t[w_1 \leftarrow r(w_1), \ldots, w_m \leftarrow r(w_m)]$ and for every $u \in pos(t')$, we put $r'(u) = r(u)$,

3. for every $1 \leq i \leq m$: $t_i = t|_{w_i}$ and for every $u \in pos(t_i)$, we let $r_i(u) = r(w_i u)$.

Also we define the *$(U,\varepsilon)$-decomposition of t and r* in the same way as the $U$-decomposition of $t$ and $r$ except that in item 1 the set of positions is given as follows: $\{w_1, \ldots, w_m\} = \{w \in pos(t) \,|\, w \in U(r)$ and for every $v \in pos(t)$ such that $v \leq_{pre} w$ and $v \neq w : v \notin U(r)\}$, i.e. now the decomposition can also take place at the root. □

In fact, we will need both kinds of decompositions. The $U$-decomposition will be needed in Theorem 4.1, which involves an induction on the size of $t$; there we may not decompose trivially at the root of $t$, since otherwise the induction hypothesis is not applicable. On the other hand, in Lemma 5.6 we have to allow also decompositions at the root.

**Observation 3.5** Let $k \geq 0$, $t, t', t_1, \ldots, t_k \in T_\Sigma(Q)$, $w_1, \ldots, w_k \in pos(t)$, $r, r', r_1, \ldots, r_k$ be runs of $M$ on $t, t', t_1, \ldots, t_k$, resp. Moreover, let $P \subseteq Q$, $q \in Q$, and $U$ be a node property with $U(r) = \{w_1, \ldots, w_m\}$. Then the following two properties are equivalent:

1. $r \in R_M^P(t,q)$ and $dec_U(t,r) = (t', r', (w_1, t_1, r_1), \ldots, (w_m, t_m, r_m))$.

2. $r' \in R_M^P(t', q)$, for every $1 \leq i \leq m$, $r_i \in R_M^P(t_i, r'(w_i))$, and $t = t'[w_1 \leftarrow t_1, \ldots, w_m \leftarrow t_m]$ and $r'(u) = r(u)$ for every $u \in pos(t')$.

Due to the commutativity of $A$, the decomposition of a run preserves the costs as follows.

**Observation 3.6** Let $t \in T_\Sigma$ and $r \in R_M(t)$, $U$ be a node property of $t$, and $dec_U(t,r) = (t', r', (w_1, t_1, r_1), \ldots, (w_m, t_m, r_m)))$. Then $c(t,r) = c(t',r') \odot c(t_1,r_1) \odot \ldots \odot c(t_m,r_m)$. $\square$

Thus, if in particular $t = \sigma(t_1, \ldots, t_k)$ and $U(r) = \{1, \ldots, k\}$, then $t' = \sigma(r(1), \ldots, r(k))$ and $c(t,r) = \delta_\sigma(r(1), \ldots, r(k), r(\varepsilon)) \odot c(t_1,r_1) \odot \ldots \odot c(t_k,r_k)$.

**Definition 3.7** $M$ is called *final state normalized* if $F$ is a singleton and, for every $k \geq 0$, $\sigma \in \Sigma^{(k)}$, $q_1, \ldots, q_k, q \in Q$, if there is an $i$ with $1 \leq i \leq k$ and $q_i \in F$, then $\delta_\sigma(q_1, \ldots, q_k, q) = 0$. $\square$

Note that, in a final state normalized bottom-up tree automaton, the unique final state is a sink state (in the usual sense as in physics).

**Lemma 3.8** For every bottom-up weigthed tree automaton $M$ there is an equivalent final state normalized bottom-up weighted tree automaton $M'$. Moreover, $M'$ has one more state than $M$.

**Proof:** Let $M = (Q, \Sigma, \delta, F)$ with costs in $A$. Construct the bottom-up weighted tree automaton $M' = (Q', \Sigma, \delta', \{q_f\})$ over $A$ as follows:

- $Q' = Q \cup \{q_f\}$ where $q_f$ is a new state,
- for every $k \geq 0$, $\sigma \in \Sigma^{(k)}$, $q_1, \ldots, q_k, q \in Q'$,

$$\delta'_\sigma(q_1, \ldots, q_k, q) = \begin{cases} \delta_\sigma(q_1, \ldots, q_k, q) & \text{if } q_1, \ldots, q_k, q \in Q \\ \sum_{p \in F} \delta_\sigma(q_1, \ldots, q_k, p) & \text{if } q_1, \ldots, q_k \in Q, q = q_f \\ 0 & \text{otherwise} \end{cases}$$

Note that, for every $r \in R_M(t,q)$, $c_M(t,r) = c_{M'}(t,r)$. Now let $t = \sigma(t_1, \ldots, t_k) \in T_\Sigma$. Using the decomposition $U(r) = \{1, \ldots, k\}$ we can calculate as follows:

$$\begin{aligned}
& (S_M, t) \\
= \; & \textstyle\sum_{q \in F} \sum_{r \in R_M(t,q)} c_M(t,r) \\
= \; & \textstyle\sum_{q \in F} \sum_{q_1, \ldots, q_k \in Q} \sum_{\substack{\forall 1 \leq i \leq k: \\ r_i \in R_M(t_i, q_i)}} \delta_\sigma(q_1, \ldots, q_k, q) \odot \prod_{i=1}^k c_M(t_i, r_i) \\
= \; & \textstyle\sum_{q_1, \ldots, q_k \in Q} \sum_{\substack{\forall 1 \leq i \leq k: \\ r_i \in R_M(t_i, q_i)}} \left( \sum_{q \in F} \delta_\sigma(q_1, \ldots, q_k, q) \right) \odot \prod_{i=1}^k c_M(t_i, r_i) \\
=^* \; & \textstyle\sum_{q_1, \ldots, q_k \in Q} \sum_{\substack{\forall 1 \leq i \leq k: \\ r_i \in R_{M'}(t_i, q_i)}} \delta'_\sigma(q_1, \ldots, q_k, q_f) \odot \prod_{i=1}^k c_{M'}(t_i, r_i) \\
= \; & \textstyle\sum_{r \in R_{M'}(t, q_f)} c_{M'}(t,r) \\
= \; & (S_{M'}, t)
\end{aligned}$$

At * we have used the fact that, for every $t \in T_\Sigma$ and $q \in Q$, $R_M(t,q) \subset R_{M'}(t,q)$. It is also obvious that, for every $r \in R_{M'}(t,q) - R_M(t,q)$, we have $c_{M'}(t,r) = 0$, because in such a run some non-root position is labeled by $q_f$. $\square$

**Definition 3.9** Let $\alpha \in \Sigma^{(0)}$. We define the set $I_\alpha = \{p \in Q \mid \delta_\alpha((), p) \neq 0\}$ and call its elements *initial $\alpha$-states.*

**Definition 3.10** Let $\alpha \in \Sigma^{(0)}$. $M$ is called *initial state $\alpha$-normalized* if there is exactly one initial $\alpha$-state $q \in Q$ and if $q$ is the initial $\alpha$-state, then $\delta_\alpha((), q) = 1$.

**Lemma 3.11** Let $\alpha \in \Sigma^{(0)}$ and $S_M$ be $\alpha$-unitary. Then there is an initial state $\alpha$-normalized bottom-up weigthed tree automaton $M'$ over $A$ such that $S_M = S_{M'}$. Moreover, $M'$ can be chosen to have one more state than $M$.

**Proof.** Construct the bottom-up weigthed tree automaton $M' = (Q', \Sigma, \delta', F')$ by $Q' = Q \cup \{q_0\}$ and $F' = F \cup \{q_0\}$ where $q_0$ is a new state, and $\delta'$ is defined as follows:

- for every $k \geq 0$, $\sigma \in \Sigma^{(k)}$ with $\sigma \neq \alpha$, $q_1, \ldots, q_k \in Q'$, and $q \in Q$, we define

$$\delta'_\sigma(q_1,\ldots,q_k,q) =$$
$$\sum\{(\prod_{\substack{1\le i\le k\\ q_i=q_0}} \delta_\alpha((),p_i))\odot\delta_\sigma(p_1,\ldots,p_k,q) \mid p_1,\ldots,p_k\in Q, p_i=q_i \text{ if } q_i\in Q\}$$

- for every $k\ge 0$, $\sigma\in\Sigma^{(k)}$ with $\sigma\neq\alpha$, and $q_1,\ldots,q_k\in Q'$, we define $\delta'_\sigma(q_1,\ldots,q_k,q_0)=0$
- for every $q\in Q$, $\delta'_\alpha((),q)=0$, and
- $\delta'_\alpha((),q_0)=1$.

Clearly, $M'$ is initial state $\alpha$-normalized where $q_0$ is the initial $\alpha$-state. Also note that for every tree $t\in T_\Sigma$, run $r$ of $M'$ on $t$, and node $w\in pos(t)$, if $lab(t,w)\neq\alpha$ and $r(w)=q_0$, then $c_{M'}(t,r)=0$.

We show that $M$ and $M'$ are equivalent, i.e., for every $t\in T_\Sigma$, we claim that $(S_M,t)=(S_{M'},t)$. First let $t=\alpha$. Then we can calculate as follows:

$$
\begin{aligned}
& (S_{M'},\alpha)\\
= \;& \textstyle\sum_{q'\in F'}\sum_{r'\in R_{M'}(\alpha,q')} c_{M'}(\alpha,r')\\
= \;& \textstyle\sum_{r'\in R_{M'}(\alpha,q_0)} c_{M'}(\alpha,r') \oplus \sum_{q\in F}\sum_{r'\in R_{M'}(\alpha,q)} c_{M'}(\alpha,r')\\
= \;& \textstyle\sum_{r'\in R_{M'}(\alpha,q_0)} c_{M'}(\alpha,r')\\
& \text{(for every } r'\in R_{M'}(\alpha,q)\text{, we have that } r'(\varepsilon)=q \text{ and thus, if } q\in F\text{, then by definition}\\
& \text{of } \delta'_\alpha,\ c_{M'}(\alpha,r',\varepsilon)=0)\\
= \;& c_{M'}(\alpha,r',\varepsilon)\\
& \text{(where } r'(\varepsilon)=q_0)\\
= \;& \delta'_\alpha((),r'(\varepsilon))\\
= \;& \delta'_\alpha((),q_0)\\
= \;& 1
\end{aligned}
$$

Next let $t\neq\alpha$. We define the mapping $\nu: R_M(t,q)\to R_{M'}(t,q)$ such that, for every $r\in R_M(t,q)$ and $w\in pos(t)$, we have

$$(\nu(r))(w) = \begin{cases} r(w) & \text{if } lab(t,w)\neq\alpha\\ q_0 & \text{otherwise}\end{cases}$$

Note that $\nu$ is not injective. We consider the nodes which are predecessors of $\alpha$-labeled nodes and the $\alpha$-labeled nodes themselves. Formally, we define

$$pre(\alpha)=\{w_1,\ldots,w_n\}=\{w\in pos(t)\mid \text{ there is an } i \text{ such that } wi\in pos_{\{\alpha\}}(t)\}$$

and for every $1\le j\le n$, we define

$$pos_j=\{i_{j1},\ldots,i_{jk_j}\}=\{i\mid w_ji\in pos_{\{\alpha\}}(t)\}.$$

Observe that $pos_{\{\alpha\}}(t)=\bigcup_{1\le j\le n}\{w_ji\mid i\in pos_j\}$. Also we define the closure of $pre(\alpha)$ as the set

$$cl(\alpha)=pre(\alpha)\cup pos_{\{\alpha\}}(t).$$

Now let $q\in Q$ and $r'\in R_{M'}(t,q)$ a run such that $r'(w)=q_0$ for every $w\in pos_\alpha(t)$. We calculate:

$$
\begin{aligned}
& \textstyle\sum_{r\in\nu^{-1}(r')}\prod_{w\in cl(\alpha)} c_M(t,r,w)\\
= \;& \textstyle\sum_{r\in\nu^{-1}(r')}\prod_{1\le j\le n}\prod_{i\in pos_j} c_M(t,r,w_ji)\odot c_M(t,r,w_j)\\
= \;& \textstyle\sum_{r\in\nu^{-1}(r')}\prod_{1\le j\le n}\prod_{1\le l\le k_j} \delta_\alpha((),r(w_ji_{jl}))\odot\\
& \qquad \delta_{\sigma_j}(r(w_j1),\ldots,r(w_j(i_{j1}-1)),\ r(w_ji_{j1}),\\
& \qquad\quad r(w_j(i_{j1}+1)),\ldots,r(w_j(i_{j2}-1)),\ r(w_ji_{j2})\\
& \qquad\quad \ldots,\\
& \qquad\quad r(w_j(i_{j(k_j-1)}+1)),\ldots,r(w_j(i_{jk_j}-1)),\ r(w_ji_{jk_j})\\
& \qquad\quad r(w_j(i_{jk_j}+1)),\ldots,r(w_j\rho_j))
\end{aligned}
$$

(where for every $1\le j\le n$, we assume that $lab(t,w_j)=\sigma_j\in\Sigma^{(\rho_j)}$)

$$
\begin{aligned}
= \quad & \prod_{1\le j\le n} \sum \Big\{ \prod_{1\le l\le k_j} \delta_\alpha((),p_{jl})) \odot \\
& \qquad R_{\sigma_j}(r'(w_j1),\ldots,r'(w_j(i_{j1}-1)),\ p_{j1}, \\
& \qquad\quad r'(w_j(i_{j1}+1)),\ldots,r'(w_j(i_{j2}-1)),\ p_{j2}) \\
& \qquad\quad \ldots, \\
& \qquad\quad r'(w_j(i_{j(k_j-1)}+1)),\ldots,r'(w_j(i_{jk_j}-1)),\ p_{jk_j}) \\
& \qquad\quad r'(w_j(i_{jk_j}+1)),\ldots,r'(w_j\rho_j)) \mid p_{j1},\ldots,p_{jk_j} \in Q \Big\} \\
= \quad & \prod_{1\le j\le n} \delta'_{\sigma_j}(r'(w_j1),\ldots,r'(w_j\rho_j),r'(w_j)) \\
= \quad & \prod_{1\le j\le n} c_{M'}(t,r',w_j) \\
= \quad & \prod_{1\le j\le n} \Big( c_{M'}(t,r',w_j) \odot \prod_{i\in pos_j} c_{M'}(t,r',w_ji) \Big) \\
& \text{(note that, for every } i \text{ and } j\text{, we have } c_{M'}(t,r',w_ji)=1) \\
= \quad & \prod_{w\in cl(\alpha)} c_{M'}(t,r',w)
\end{aligned}
$$

Now we obtain:

$$
\begin{aligned}
& (S_M,t) \\
= \quad & \sum_{q\in F} \sum_{r\in R_M(t,q)} c_M(t,r) \\
= \quad & \sum_{q\in F} \sum_{r\in R_M(t,q)} \prod_{w\in pos(t)} c_M(t,r,w) \\
= \quad & \sum_{q\in F} \sum_{r\in R_M(t,q)} \\
& \qquad \prod_{w\in pos(t)-cl(\alpha)} c_M(t,r,w) \ \odot\ \prod_{w\in cl(\alpha)} c_M(t,r,w) \\
=^{*1} \quad & \sum_{q\in F} \sum_{\substack{r'\in R_{M'}(t,q) \\ \forall w\in pos_{\{\alpha\}}(t): r'(w)=q_0}} \sum_{r\in \nu^{-1}(r')} \\
& \qquad \prod_{w\in pos(t)-cl(\alpha)} c_M(t,r,w) \ \odot\ \prod_{w\in cl(\alpha)} c_M(t,r,w) \\
=^{*2} \quad & \sum_{q\in F} \sum_{\substack{r'\in R_{M'}(t,q) \\ \forall w\in pos_{\{\alpha\}}(t): r'(w)=q_0}} \sum_{r\in \nu^{-1}(r')} \\
& \qquad \prod_{w\in pos(t)-cl(\alpha)} c_{M'}(t,r',w) \ \odot\ \prod_{w\in cl(\alpha)} c_M(t,r,w) \\
= \quad & \sum_{q\in F} \sum_{\substack{r'\in R_{M'}(t,q) \\ \forall w\in pos_{\{\alpha\}}(t): r'(w)=q_0}} \prod_{w\in pos(t)-cl(\alpha)} c_{M'}(t,r',w) \odot \\
& \qquad \sum_{r\in \nu^{-1}(r')} \prod_{w\in cl(\alpha)} c_M(t,r,w) \\
= \quad & \sum_{q\in F} \sum_{\substack{r'\in R_{M'}(t,q) \\ \forall w\in pos_{\{\alpha\}}(t): r'(w)=q_0}} \prod_{w\in pos(t)-cl(\alpha)} c_{M'}(t,r',w) \odot \\
& \qquad \prod_{w\in cl(\alpha)} c_{M'}(t,r',w) \\
& \text{(by the previous calculation)} \\
= \quad & \sum_{q\in F} \sum_{\substack{r'\in R_{M'}(t,q) \\ \forall w\in pos_{\{\alpha\}}(t): r'(w)=q_0}} \prod_{w\in pos(t)} c_{M'}(t,r',w) \\
= \quad & \sum_{q\in F} \sum_{r'\in R_{M'}(t,q)} \prod_{w\in pos(t)} c_{M'}(t,r',w) \\
= \quad & \sum_{q\in F} \sum_{r'\in R_{M'}(t,q)} c_{M'}(t,r') \\
= \quad & \sum_{r'\in R_{M'}(t,q_0)} c_{M'}(t,r') \ \oplus\ \sum_{q\in F} \sum_{r'\in R_{M'}(t,q)} c_{M'}(t,r') \\
& \text{(because for every } r'\in R_{M'}(t,q_0) \text{ and } t\neq\alpha\text{, the value } c_{M'}(t,r')=0 \text{ by definition of } M')
\end{aligned}
$$

$$
\begin{array}{rl}
= & \sum_{q'\in F'} \sum_{r'\in R_{M'}(t,q')} c_{M'}(t,r') \\
= & (S_{M'},t)
\end{array}
$$

At $*1$ we have used the fact that $\{\nu^{-1}(r') \mid r' \in R_{M'}(t,q), \forall w \in pos_{\{\alpha\}}(t) : r'(w) = q_0\}$ is a partitioning of $R_M(t,q)$. At $*2$ we have used the following fact: for every $r' \in R_{M'}(t,q)$, $r \in \nu^{-1}(r')$, and $w \in pos(t) - cl(\alpha)$ with $lab(t,w) = \sigma$ for some $\sigma \in \Sigma^{(k)}$, $k \geq 0$, we have $\sigma \neq \alpha$ and::

$$
\begin{array}{rll}
 & c_M(t,r,w) & \\
= & \delta_\sigma(r(w1),\ldots,r(wk),r(w)) & \\
= & \delta_\sigma(r'(w1),\ldots,r'(wk),r'(w)) & \\
= & \delta'_\sigma(r'(w1),\ldots,r'(wk),r'(w)) & \text{(because } r'(w1),\ldots,r'(wk),r'(w) \in Q) \\
= & c_{M'}(t,r',w) &
\end{array}
$$

□

**Definition 3.12** Let $\alpha \in \Sigma^{(0)}$ and $S_M$ be $\alpha$-unitary. $M$ is *$\alpha$-normalized* if

- $M$ is final state normalized with final state $q_f$ and
- $I_\alpha = \{q_f, q_0\}$ such that $q_0 \notin F$ and $\delta_\alpha((), q_f) = \delta_\alpha((), q_0) = 1$.

**Lemma 3.13** Let $\alpha \in \Sigma^{(0)}$ and $S_M$ $\alpha$-unitary. Then there is an $\alpha$-normalized bottom-up weighted tree automaton $M'$ over $A$ such that $S_M = S_{M'}$.

**Proof.** To a given bottom-up weighted tree automaton $M$, we first apply the construction of Lemma 3.11 thereby obtaining $M'$; $M'$ is initial state $\alpha$-normalized. Next we apply to $M'$ the construction of Lemma 3.8 and obtain $M''$; $M''$ is final state normalized, but the new final state can read $\alpha$ with cost 1. In fact, $M''$ satisfies the requirements. □

# 4 From automata to rational series

Kleene's result says that the classes of recognizable languages and of rational languages are equal. We will show that the same is true for recognizable and rational tree series if the semiring $A$ is commutative and idempotent. First let us prove that every recognizable tree series is rational.

**Theorem 4.1** Let $A$ be commutative and idempotent. Then $A^{rec}\langle\langle T_\Sigma\rangle\rangle \subseteq \bigcup_{Q \text{ finite set}} A^{rat}\langle\langle T_\Sigma(Q)\rangle\rangle$.

**Proof.** Let $M = (Q, \Sigma, \delta, F)$ be a bottom-up weighted tree automaton. We will prove that $S_M \in A^{rat}\langle\langle T_\Sigma(Q)\rangle\rangle$. For every $P \subseteq Q$ and $q \in Q$ we define the tree series $S_M(P,q) \in A\langle\langle T_\Sigma(Q)\rangle\rangle$ such that for every $t \in T_\Sigma(Q)$,

$$(S_M(P,q),t) = \sum_{r\in R^P_M(t,q)} c(t,r).$$

Then, obviously, for every $t \in T_\Sigma(Q)$, $(S_M,t) = \sum_{q\in F}(S_M(Q,q),t)$. For every $P \subseteq Q$ and $q \in Q$, we will prove by induction on the number of elements in $P$ that the tree series $S_M(P,q)$ is rational.

For the induction base (i.e., $P = \emptyset$), let $r_{\varepsilon,q} : \{\varepsilon\} \to Q$ be the run such that $r_{\varepsilon,q}(\varepsilon) = q$. Also for every $k \geq 0$, $\sigma \in \Sigma^{(k)}$, and $q_1,\ldots,q_k \in Q$, let $r^\sigma_{q_1\ldots q_k,q} : pos(\sigma(q_1,\ldots,q_k)) \to Q$ be the run of $M$ on $\sigma(q_1,\ldots,q_k)$ using $\emptyset$ such that $r^\sigma_{q_1\ldots q_k,q}(\varepsilon) = q$. Then by definition we have

$$R^\emptyset_M(t,q) = \begin{cases} \{r_{\varepsilon,q}\} & \text{if } t = q \\ \{r^\sigma_{q_1\ldots q_k,q}\} & \text{if } t = \sigma(q_1,\ldots,q_k) \text{ for some } q_1,\ldots,q_k \in Q \\ \emptyset & \text{otherwise} \end{cases}$$

Thus $supp(S_M(\emptyset,q)) \subseteq \{q\} \cup \Sigma(Q)$ where $\Sigma(Q) = \{\sigma(q_1,\ldots,q_k) \mid k \geq 0, q_1,\ldots,q_k \in Q, \sigma \in \Sigma^{(k)}\}$ and if $t = q$, then $c(t, r_{\varepsilon,q}) = 1$ and, if $t = \sigma(q_1,\ldots,q_k)$, then $c(t, r^\sigma_{q_1\ldots q_k,q}) = \delta_\sigma(q_1,\ldots,q_k,q)$. And thus, $S_M(\emptyset,q) = 1\,q \oplus \sum_{\sigma(q_1,\ldots,q_k)\in\Sigma(Q)} \delta_\sigma(q_1,\ldots,q_k,q)\,\sigma(q_1,\ldots,q_k)$, which is a finite sum of monomials, and hence, by Observation 2.27, $S_M(\emptyset,q)$ is rational.

For the induction step, we assume that, for every $q \in Q$, $S_M(P,q)$ is rational and let $p \in Q - P$. Then, we will prove that for every $q \in Q$, also $S_M(P \cup \{p\}, q)$ is rational. This statement follows from the claim:

$$(*) \quad \text{for every } t \in T_\Sigma(Q)\text{: } (S_M(P \cup \{p\}, q), t) = (S_M(P, q) \cdot_p S_M(P, p)^{*p}, t)$$

which shows that $S_M(P \cup \{p\}, q)$ is built up from rational tree series by rational operations; note that $S_M(P, p)$ is $p$-unitary. Next we will prove claim $(*)$ by induction on the size of $t$. If $t \in \Sigma \cup Q$, our claim is clear. Now let $size(t) \geq 2$. Note that, for every $q \in Q$ (and thus, in particular, also for $q = p$), $R_M^P(t, q) \subseteq R_M^{P \cup \{p\}}(t, q)$. We calculate

$$\begin{aligned}
& (S_M(P \cup \{p\}, q), t) \\
= \quad & \textstyle\sum_{r \in R_M^{P \cup \{p\}}(t,q)} c(t, r) \\
=^* \quad & \sum_{\substack{r' \in R_M^P(s,q) \\ r_i \in R_M^{P \cup \{p\}}(u_i, p) \\ t = s[p \leftarrow (u_1, \ldots, u_k)]}} c(s, r') \odot c(u_1, r_1) \odot \ldots \odot c(u_k, r_k) \\
& \text{(by Observations 3.5 and 3.6; note that by definition of } dec_U(t, r) \text{ we have } s \neq p) \\
= \quad & \sum_{\substack{r' \in R_M^P(s,q) \\ t = s[p \leftarrow (u_1, \ldots, u_k)]}} c(s, r') \odot \Big(\textstyle\sum_{r_1 \in R_M^{P \cup \{p\}}(u_1, p)} c(u_1, r_1)\Big) \odot \ldots \odot \Big(\textstyle\sum_{r_k \in R_M^{P \cup \{p\}}(u_k, p)} c(u_k, r_k)\Big) \\
& \text{(by distributivity of } A) \\
= \quad & \sum_{\substack{r' \in R_M^P(s,q) \\ t = s[p \leftarrow (u_1, \ldots, u_k)]}} c(s, r') \odot (S_M(P \cup \{p\}, p), u_1) \odot \ldots \odot (S_M(P \cup \{p\}, p), u_k) \\
& \text{(by definition of } (S_M(P \cup \{p\}, p), u_i)) \\
= \quad & \sum_{\substack{r' \in R_M^P(s,q) \\ t = s[p \leftarrow (u_1, \ldots, u_k)]}} c(s, r') \odot (S_M(P, p) \cdot_p S_M(P, p)^{*p}, u_1) \odot \ldots \odot (S_M(P, p) \cdot_p S_M(P, p)^{*p}, u_k) \\
& \text{(by induction hypothesis on } u_i\text{'s; note that, since } s \neq p \text{, we have } size(u_i) < size(t)) \\
= \quad & \sum_{\substack{s, u_1, \ldots, u_k \in T_\Sigma(Q) \\ t = s[p \leftarrow (u_1, \ldots, u_k)]}} \Big(\textstyle\sum_{r' \in R_M^P(s,q)} c(s, r')\Big) \odot (S_M(P, p)^{*p}, u_1) \odot \ldots \odot (S_M(P, p)^{*p}, u_k) \\
& \text{(by distributivity and Lemma 2.18)} \\
= \quad & \sum_{\substack{s, u_1, \ldots, u_k \in T_\Sigma(Q) \\ t = s[p \leftarrow (u_1, \ldots, u_k)]}} (S_M(P, q), s) \odot (S_M(P, p)^{*p}, u_1) \odot \ldots \odot (S_M(P, p)^{*p}, u_k) \\
= \quad & (S_M(P, q) \cdot_p S_M(P, p)^{*p}, t)
\end{aligned}$$

At equation * we have used the fact that $t$ and $r$ are uniquely decomposed by $dec_U(t, r) = (s, r', (w_1, u_1, r_1), \ldots, (w_k, u_k, r_k))$, where $U(r) = \{w \in pos(t) \,|\, r(w) = p\}$ is a node property of $t$. $\square$

Recall, e.g., from p.14 of [Per90] that in the string case one has to take care about the initial states of paths $q_i \to p_1 \to p_2 \to \ldots \to p_n \to q_j$, and form the set $X^P_{q_i, q_j}$ of all labels of such paths where $\{p_1, \ldots, p_n\} \subseteq P$. This is necessary because two paths $q_i \to^* q_j$ and $q_k \to^* q_l$ can be put together only if $q_j = q_k$. In the tree case however, the checking whether $q_j = q_k$ is done by means of the $q_k$-concatenation of trees. Thus, in claim (*) we were able to drop the first subscript-parameter from $X^P_{q_i, q_j}$.

We note that the concatenation in the previous theorem only uses symbols from $Q$, as in the classical case of tree languages.

# 5 Rational constructions

Here we prove that $A^{rat}\langle\langle T_\Sigma \rangle\rangle \subseteq A^{rec}\langle\langle T_\Sigma \rangle\rangle$. Since $A^{rat}\langle\langle T_\Sigma \rangle\rangle$ is the smallest class containing the polynomials and being closed under $\oplus$, $\alpha$-concatenation, and Kleene-star, we only have to prove that $A^{rec}\langle\langle T_\Sigma \rangle\rangle$ also contains the polynomials and is closed under these operations.

**Lemma 5.1** Let $A$ be a commutative semiring. Then $A^{rec}\langle\langle T_\Sigma \rangle\rangle$ contains all tree series of the form $1\,\beta$ with $\beta \in \Sigma^{(0)}$.

**Proof.** Construct the bottom-up weighted tree automaton $M = (Q, \Sigma, \delta, F)$ over $A$ as follows: $Q = F = \{[\beta]\}$ and for every $k \geq 0$, $\sigma \in \Sigma^{(k)}$, we let

$$\delta_\sigma(\underbrace{[\beta], \ldots, [\beta]}_{k}, [\beta]) = \begin{cases} 1 & \text{if } \sigma = \beta \\ 0 & \text{otherwise} \end{cases}$$

Clearly, $S_M = 1\beta$. □

**Lemma 5.2** Let $A$ be a commutative semiring. Then $A^{rec}\langle\langle T_\Sigma \rangle\rangle$ is closed under top-concatenation.

**Proof.** Let $\sigma \in \Sigma^{(k)}$ with $k \geq 1$ and let $M_i = (Q_i, \Sigma, \delta_i, F_i)$ be bottom-up weighted tree automata such that for every $i, j$, if $i \neq j$, then $Q_i \cap Q_j = \emptyset$. Construct the bottom-up weighted tree automaton $M = (Q, \Sigma, \delta, F)$ by defining $Q = \{q_f\} \cup \bigcup_{1 \leq i \leq k} Q_i$ where $q_f$ is a new state, $F = \{q_f\}$, and for every $l \geq 0$, $\gamma \in \Sigma^{(l)}$, $q_1, \ldots, q_l, q \in Q$,

$$\delta_\gamma(q_1, \ldots, q_l, q) = \begin{cases} (\delta_i)_\gamma(q_1, \ldots, q_l, q) & \text{if } q, q_1, \ldots, q_l \in Q_i \\ 1 & \text{if } \gamma = \sigma, l = k \text{ and for every } 1 \leq i \leq l, q_i \in F_i \text{ and } q = q_f \\ 0 & \text{otherwise} \end{cases}$$

Then, for every $t = \sigma(t_1, \ldots, t_k) \in T_\Sigma$,

$$\begin{aligned}
& (S_M, t) \\
= \; & \textstyle\sum_{q \in F} \sum_{r \in R_M(t,q)} c_M(t, r) \\
= \; & \textstyle\sum_{r \in R_M(t,q_f)} c_M(t, r) \\
= \; & \textstyle\sum_{\substack{r_1 \in R_M(t_1, q_{f_1}) \\ q_{f_1} \in F_1}} \cdots \sum_{\substack{r_k \in R_M(t_k, q_{f_k}) \\ q_{f_k} \in F_k}} c_M(t_1, r_1) \odot \ldots \odot c_M(t_k, r_k) \\
= \; & \left( \textstyle\sum_{\substack{r_1 \in R_M(t_1, q_{f_1}) \\ q_{f_1} \in F_1}} c_M(t_1, r_1) \right) \odot \ldots \odot \left( \sum_{\substack{r_k \in R_M(t_k, q_{f_k}) \\ q_{f_k} \in F_k}} c_M(t_k, r_k) \right) \\
= \; & \left( \textstyle\sum_{\substack{r_1 \in R_{M_1}(t_1, q_{f_1}) \\ q_{f_1} \in F_1}} c_{M_1}(t_1, r_1) \right) \odot \ldots \odot \left( \sum_{\substack{r_k \in R_{M_k}(t_k, q_{f_k}) \\ q_{f_k} \in F_k}} c_{M_k}(t_k, r_k) \right) \\
= \; & (S_{M_1}, t_1) \odot \ldots \odot (S_{M_k}, t_k) \\
= \; & (top_\sigma(S_{M_1}, \ldots, S_{M_k}), t)
\end{aligned}$$

For every $t \in T_\Sigma$ such that $lab(t, \varepsilon) \neq \sigma$, we have $(S_M, t) = 0$, because $R_M(t, q_f)$ contains only runs with cost 0. Thus, in total, we obtain $S_M = top_\sigma(S_{M_1}, \ldots, S_{M_k})$. □

**Lemma 5.3** Let $A$ be a commutative semiring. Then $A^{rec}\langle\langle T_\Sigma \rangle\rangle$ is closed under multiplication with a coefficient.

**Proof.** Let $M = (Q, \Sigma, \delta, F)$ be any bottom-up tree automaton, and let $a \in A$. By Lemma 3.8 we can assume that $M$ is final state normalized with $F = \{q_f\}$. Construct the bottom-up weighted tree automaton $M' = (Q, \Sigma, \delta', F)$ over $A$ such that for every $k \geq 0$, $\sigma \in \Sigma^{(k)}$, and $q, q_1, \ldots, q_k \in Q$:

$$\delta'_\sigma(q_1, \ldots, q_k, q) = \begin{cases} a \odot \delta_\sigma(q_1, \ldots, q_k, q) & \text{if } q = q_f \\ \delta_\sigma(q_1, \ldots, q_k, q) & \text{otherwise} \end{cases}$$

Clearly, $S_{M'} = a \odot S_M$. □

**Lemma 5.4** Let $A$ be a commutative semiring. $A^{rec}\langle\langle T_\Sigma \rangle\rangle$ is closed under sum.

**Proof.** Let $M_1 = (Q_1, \Sigma, \delta_1, F_1)$ and $M_2 = (Q_2, \Sigma, \delta_2, F_2)$ be two bottom-up weighted tree automata such that $Q_1 \cap Q_2 = \emptyset$. Construct the bottom-up weighted tree automaton $M = (Q_1 \cup Q_2, \Sigma, \delta, F)$ over $A$ such that $F = F_1 \cup F_2$ and for every $k \geq 0$, $\sigma \in \Sigma^{(k)}$, $q, q_1, \ldots, q_k \in Q_1 \cup Q_2$:

$$\delta'_\sigma(q_1,\ldots,q_k,q) = \begin{cases} (\delta_1)_\sigma(q_1,\ldots,q_k,q) & \text{if } q,q_1,\ldots,q_k \in Q_1 \\ (\delta_2)_\sigma(q_1,\ldots,q_k,q) & \text{if } q,q_1,\ldots,q_k \in Q_2 \\ 0 & \text{otherwise} \end{cases}$$

Then

$$\begin{aligned} (S_M,t) &= \textstyle\sum_{q\in F}\sum_{r\in R_M(t,q)} c_M(t,r) \\ &= \textstyle\sum_{q\in F_1}\sum_{r_1\in R_{M_1}(t,q)} c_{M_1}(t,r_1) \;\oplus\; \sum_{q\in F_2}\sum_{r_2\in R_{M_2}(t,q)} c_{M_2}(t,r_2) \\ &= (S_{M_1},t)\oplus(S_{M_2},t) \\ &= (S_{M_1}\oplus S_{M_2},t). \end{aligned}$$

□

**Corollary 5.5** Let $A$ be an arbitrary semiring. Then $A^{rec}\langle\langle T_\Sigma\rangle\rangle$ contains all polynomials.

**Proof.** Since every polynomial has the form $a_1\,t_1\oplus\ldots\oplus a_n\,t_n$, the statement follows immediately from Lemma 5.1, 5.3, and 5.4. □

**Lemma 5.6** Let $A$ be a commutative semiring. Then, if $S_{M_1}, S_{M_2} \in A^{rec}\langle\langle T_\Sigma\rangle\rangle$ and $\alpha\in\Sigma^{(0)}$, then $S_{M_2}\cdot_\alpha S_{M_1}\in A^{rec}\langle\langle T_\Sigma\rangle\rangle$. Thus $A^{rec}\langle\langle T_\Sigma\rangle\rangle$ is closed under $\alpha$-concatenation.

**Proof.** Let $M_1=(Q_1,\Sigma,\delta_1,F_1)$ and $M_2=(Q_2,\Sigma,\delta_2,F_2)$ be two bottom-up weighted tree automata such that $Q_1\cap Q_2=\emptyset$. By Lemma 3.8 we may assume that $M_1$ is final state normalized with $F_1=\{q_f\}$. Construct the bottom-up weighted tree automaton $M=(Q,\Sigma,\delta,F)$ over $A$ as follows: $Q=(Q_1\cup Q_2)-\{q_f\}$, $F=F_2$, and for every $k\geq 0$, $\sigma\in\Sigma^{(k)}$, $q_1,\ldots,q_k,q\in Q$, define

$$\delta_\sigma(q_1,\ldots,q_k,q) = \begin{cases} (\delta_1)_\sigma(q_1,\ldots,q_k,q) & \text{if } q_1,\ldots,q_k,q\in Q_1 \\ (\delta_1)_\sigma(q_1,\ldots,q_k,q_f)\odot(\delta_2)_\alpha((),q) & \text{if } q_1,\ldots,q_k\in Q_1, q\in Q_2 \\ (\delta_2)_\sigma(q_1,\ldots,q_k,q) & \text{if } q_1,\ldots,q_k,q\in Q_2, \sigma\neq\alpha \\ 0 & \text{otherwise} \end{cases}$$

Now we prove that, for every $t\in T_\Sigma$, $(S_M,t)=(S_{M_2}\cdot_\alpha S_{M_1},t)$. We use the technique of decomposing a run $r$ of $M$. More precisely, for every $r\in R_M(t,q)$, we let

$$U(r)=\{w\in pos(t)\,|\,r(w)\in Q_2 \text{ and if } rk(lab(t,w))=k, \text{ then } r(w1),\ldots,r(wk)\in Q_1\}$$

be a node property of $t$ and we will decompose $r$ according to this node property (note that we have to take the $(U,\varepsilon)$-decomposition here, because possibly $\alpha\in supp(S_{M_2})$). Then

$$\begin{aligned} & (S_M,t) \\ =\; & \textstyle\sum_{q\in F_2}\sum_{r\in R_M(t,q)} c_M(t,r) \\ =\; & \textstyle\sum_{q\in F_2}\sum_{\substack{s,u_1,\ldots,u_m\in T_\Sigma\\ t=s[\alpha\leftarrow(u_1,\ldots,u_m)]}} \\ & \qquad \textstyle\sum_{r'\in R_{M_2}(s,q)}\sum_{r'_1\in R_{M_1}(u_1,q_f)}\cdots\sum_{r'_m\in R_{M_1}(u_m,q_f)} c_{M_2}(s,r')\odot(\prod_{i=1}^m c_{M_1}(u_i,r'_i)) \\ & \text{where } dec_{U,\varepsilon}(t,r)=(s,r',(w_1,u_1,r_1),\ldots,(w_m,u_m,r_m)), \\ & \text{and } r'_i \text{ is obtained from } r_i \text{ by letting } r'_i(\varepsilon)=q_f \\ =\; & \textstyle\sum_{\substack{s,u_1,\ldots,u_m\in T_\Sigma\\ t=s[\alpha\leftarrow(u_1,\ldots,u_m)]}} (\sum_{q\in F_2}\sum_{r'\in R_{M_2}(s,q)} c_{M_2}(s,r')) \\ & \qquad \textstyle\odot(\sum_{r'_1\in R_{M_1}(u_1,q_f)} c_{M_1}(u_1,r'_1))\odot\ldots\odot(\sum_{r'_m\in R_{M_1}(u_r,q_f)} c_{M_1}(u_m,r'_m)) \\ & \text{(by distributivity of } A) \\ =\; & \textstyle\sum_{\substack{s,u_1,\ldots,u_m\in T_\Sigma\\ t=s[\alpha\leftarrow(u_1,\ldots,u_m)]}} (S_{M_2},s)\odot(S_{M_1},u_1)\odot\ldots\odot(S_{M_1},u_m) \\ =\; & (S_{M_2}\cdot_\alpha S_{M_1},t). \end{aligned}$$

□

Now we turn to the recognizability of $(S_M)^{*\alpha}$ provided that $S_M$ is recognizable and $\alpha$-unitary. For this purpose we need some more auxiliary definitions. In particular, we have to decompose a tree $t$ with its run $r$ not only once but repeatedly according to some node property $U$. This leads to the notion of complete $U$-decomposition. Note that here we use both types, the $U$-decomposition and the $(U,\varepsilon)$-decomposition (cf. Figure 1 for an example of a complete decomposition; cf. also Example 5.8).

**Definition 5.7** Let $m \geq 1$, $t_1, \ldots, t_m \in T_\Sigma$, and $r_i \in R_M(t_i)$ for every $1 \leq i \leq m$. Moreover, let $U = (U_t \mid t \in T_\Sigma(Q))$ be a family of node properties. Define the *complete $U$-decomposition of* $((t_1,r_1),\ldots,(t_m,r_m))$, denoted by $cdec_{U,M}((t_1,r_1),\ldots,(t_m,r_m))$, recursively as follows:

if for every $1 \leq j \leq m$, we have that $t_j = \alpha$, then

$$cdec_U((t_1,r_1),\ldots,(t_m,r_m)) = \varepsilon$$

and otherwise

$$cdec_U((t_1,r_1),\ldots,(t_m,r_m)) =$$
$$((t_{1,0},r_{1,0}),\ldots,(t_{m,0},r_{m,0}))\ \ cdec_U(\underbrace{(t_{1,1},r_{1,1}),\ldots,(t_{1,k_1},r_{1,k_1})}_{\text{obtained from } t_1 \text{ and } r_1},\ \ldots,\ \underbrace{(t_{m,1},r_{m,1}),\ldots,(t_{m,k_m},r_{m,k_m})}_{\text{obtained from } t_m \text{ and } r_m})$$

where for every $1 \leq i \leq m$,

- if $t_i \neq \alpha$, then $dec_{U_{t_i}}(t_i,r_i) = (t_{i,0}, r_{i,0}, (w_{i,1}, t_{i,1}, r_{i,1}), \ldots, (w_{i,k_i}, t_{i,k_i}, r_{i,k_i}))$ and
- if $t_i = \alpha$, then $dec_{U_{t_i},\varepsilon}(t_i,r_i) = (t_{i,0}, r_{i,0}, (w_{i,1}, t_{i,1}, r_{i,1}), \ldots, (w_{i,k_i}, t_{i,k_i}, r_{i,k_i}))$ (note that $k_i = 1$ and $t_{i,1} = \alpha$)

**Example 5.8** Let $S = \{\sigma^{(2)}, \gamma^{(1)}, \delta^{(1)}, \alpha^{(0)}, \beta^{(0)}\}$ and let $Q = \{f, p_1, p_2, p_3, p_4\}$. Fig. 1 (a) shows a tree $t$ and a run $r$ on $t$, where at every position $w$, the state $r(w)$ is directly shown in the label at $w$. Moreover, let $U(t,r) = \{w \in pos(t) \mid r(w) = f\}$. In Fig. 1 (b) the complete decomposition of $(t,r)$ is shown. It is computed as follows where we only show the tree part and indicate the run part by "$\ldots$":

$$\begin{aligned}
& cdec_U((t,\ldots)) \\
= \ & ((\sigma(\sigma(f,f),\sigma(\beta,f)),\ldots))\ cdec_U((\gamma(\sigma(\alpha,\beta)),\ldots)(\alpha,\ldots)(\beta,\ldots)(\delta(\beta),\ldots)) \\
= \ & ((\sigma(\sigma(f,f),\sigma(\beta,f)),\ldots))\ ((\gamma(f),\ldots)(f,\ldots)(\delta(\beta),\ldots))\ cdec_U((\sigma(\alpha,\beta),\ldots)(\alpha,\ldots)) \\
= \ & ((\sigma(\sigma(f,f),\sigma(\beta,f)),\ldots))\ ((\gamma(f),\ldots)(f,\ldots)(\delta(\beta),\ldots))\ ((\sigma(f,\beta),\ldots)(f,\ldots))\ cdec_U((\alpha,\ldots)(\alpha,\ldots)) \\
= \ & ((\sigma(\sigma(f,f),\sigma(\beta,f)),\ldots))\ ((\gamma(f),\ldots)(f,\ldots)(\delta(\beta),\ldots))\ ((\sigma(f,\beta),\ldots)(f,\ldots))
\end{aligned}$$

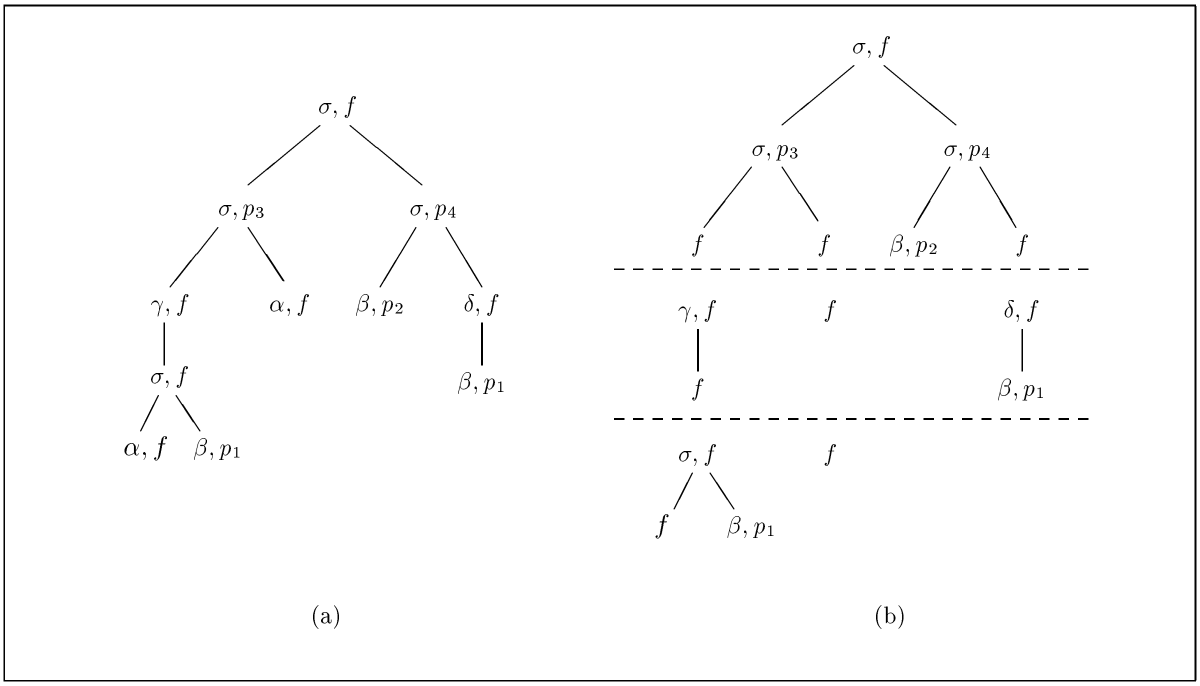


Figure 1: A tree $t$ with its run $r$ and the complete decomposition $dec_U((t,r))$.

**Lemma 5.9** Let $A$ be commutative and idempotent. Let $\alpha \in \Sigma^{(0)}$ and $S_M \in A^{rec}\langle\langle T_\Sigma\rangle\rangle$ be $\alpha$-unitary. Then $(S_M)^{*\alpha} \in A^{rec}\langle\langle T_\Sigma\rangle\rangle$. Thus $A^{rec}\langle\langle T_\Sigma\rangle\rangle$ is closed under iteration for unitary tree series.

**Proof.** By Lemma 3.13 we can assume that $M = (Q, \Sigma, \delta, F)$ is $\alpha$-normalized. Assume that $F = \{q_f\}$ and $I_\alpha = \{q_0, q_f\}$. Construct the bottom-up weighted tree automaton $M^{(*)} = (Q^{(*)}, \Sigma, \delta^{(*)}, F^{(*)})$ over $A$ where

- $Q^{(*)} = (Q - \{q_f, q_0\}) \cup F^{(*)}$ and $F^{(*)} = \{f\}$ and $f$ is a new state
- $\delta^{(*)}$ is defined as follows: for every $k \geq 0$, $\sigma \in \Sigma^{(k)}$ such that $\sigma \neq \alpha$ and every $q_1, \ldots, q_k, q \in Q^{(*)}$, define $\delta^{(*)}_\sigma(q_1, \ldots, q_k, q) = \delta_\sigma(\theta_1, \ldots, \theta_k, \theta)$ where

$$\theta_i = \begin{cases} q_0 & \text{if } q_i = f \\ q_i & \text{otherwise} \end{cases} \quad \text{and} \quad \theta = \begin{cases} q_f & \text{if } q = f \\ q & \text{otherwise} \end{cases}$$

  and

$$R^{(*)}_\alpha((), q) = \begin{cases} 1 & \text{if } q = f \\ 0 & \text{otherwise} \end{cases}$$

For every $t \in T_\Sigma$, we have to prove that $(S_{M^{(*)}}, t) = ((S_M)^*, t)$. Let $t = \alpha$. Then $(S_{M^{(*)}}, t) = (S_{M^{(*)}}, \alpha) = \sum_{r \in R_{M^{(*)}}(\alpha, f)} c_{M^{(*)}}(\alpha, r) = \delta^{(*)}_\alpha((), f) = 1$. On the other hand, since $S_M$ is $\alpha$-unitary, also $(S_M)^*$ is $\alpha$-unitary (by Definition 2.14), and thus $((S_M)^*, \alpha) = 1$.

Now let $t \in T_\Sigma - \{\alpha\}$ and $n = size_{\neg\alpha}(t)$. For every $r \in R_{M^{(*)}}(t)$, we define $U(r) = \{w \in pos(t) \mid r(w) = f\}$, a node property of $t$. Then we can prove the following Claim (*):

$$\begin{aligned}\{c_{M^{(*)}}(t, r) \mid\ & r \in R_{M^{(*)}}(t, f)\} = \\ & \{c_M(\underline{s}, \underline{r}') \odot \prod_{\substack{n-1 \geq i \geq 1 \\ 1 \leq j \leq m_i}} c_M(\underline{u}^i_j, \underline{r}^i_j) \mid x = (\underline{s}, (\underline{u}^i_j)_{\substack{n-1 \geq i \geq 1 \\ 1 \leq j \leq m_i}}) \in M_{\alpha,n}(t), \\ & \qquad \underline{r}' \in R_M(\underline{s}, q_f), \underline{r}^i_j \in R_M(\underline{u}^i_j, q_f)\}\end{aligned}$$

Proof of Claim (*), $\subseteq$: Let $r \in R_{M^{(*)}}(t, f)$ and let, for some $1 \leq \kappa \leq n$,

$$cdec_U(t, r) = ((s, r'))((u^{\kappa-1}_1, r^{\kappa-1}_1), \ldots, (u^{\kappa-1}_{m_{\kappa-1}}, r^{\kappa-1}_{m_{\kappa-1}})) \ldots ((u^1_1, r^1_1), \ldots, (u^1_{m_1}, r^1_{m_1})).$$

Then define $\underline{r}'$, $\underline{r}^i_j$, and $x$ as follows:

$$x = \begin{cases} (\underline{s}, (\underline{u}^i_j)_{\substack{\kappa-1 \geq i \geq 1 \\ 1 \leq j \leq m_i}}) & \text{if } \kappa = n \\ (\alpha, \underbrace{(\alpha), \ldots, (\alpha)}_{n-\kappa-1}, (\underline{s}), (\underline{u}^i_j)_{\substack{\kappa-1 \geq i \geq 1 \\ 1 \leq j \leq m_i}}) & \text{if } \kappa < n \end{cases}$$

where $\underline{s}$ and $\underline{u}^i_j$ are obtained from $s$ and $u^i_j$, resp., by replacing every occurrence of $f$ as a leaf by $\alpha$, and for every $w \in pos(\underline{s})$ and $v \in pos(\underline{u}^i_j)$, we define

$$\underline{r}'(w) = \begin{cases} q_f & \text{if } w = \varepsilon \\ q_0 & \text{if } lab(s, w) = f \\ r'(w) & \text{otherwise} \end{cases} \quad \text{and} \quad \underline{r}^i_j(v) = \begin{cases} q_f & \text{if } v = \varepsilon \\ q_0 & \text{if } lab(u^i_j, v) = f \\ r^i_j(v) & \text{otherwise} \end{cases}$$

Due to the construction of $M^{(*)}$ and the definition of $U$, it is clear that $\underline{r}' \in R_M(\underline{s}, q_f)$ and $\underline{r}^i_j \in R_M(\underline{u}^i_j, q_f)$. Then, by using Observation 3.6, we can compute as follows:

$$c_{M^{(*)}}(t, r) = c_{M^{(*)}}(s, r') \odot \prod_{\substack{\kappa-1 \geq i \geq 1 \\ 1 \leq j \leq m_i}} c_{M^{(*)}}(u^i_j, r^i_j) = c_M(\underline{s}, \underline{r}') \odot \prod_{\substack{\kappa-1 \geq i \geq 1 \\ 1 \leq j \leq m_i}} c_M(\underline{u}^i_j, \underline{r}^i_j) =$$

$$= c_M(\alpha, r_{q_f}) \odot \underbrace{c_M(\alpha, r_{q_f}) \odot \ldots \odot c_M(\alpha, r_{q_f})}_{n-\kappa-1} \odot c_M(\underline{s}, \underline{r}') \odot \prod_{\substack{\kappa-1 \geq i \geq 1 \\ 1 \leq j \leq m_i}} c_M(\underline{u}_j^i, \underline{r}_j^i)$$

where $r_{q_f}(\varepsilon) = q_f$. Note that $c_M(\alpha, r_{q_f}) = 1$ because $S_M$ is $\alpha$-unitary and $q_f$ is the only final state of $M$.

$\supseteq$: Let $x = (\underline{s}, (\underline{u}_j^i)_{\substack{n-1 \geq i \geq 1 \\ 1 \leq j \leq m_i}}) \in M_{\alpha,n}(t)$, and $\underline{r}' \in R_M(\underline{s}, q_f)$ and $\underline{r}_j^i \in R_M(\underline{u}_j^i, q_f)$. Let $w_j^i \in pos(t)$ denote the root of $u_j^i$ with respect to $t$. We construct a run $r$ of $M^{(*)}$ on $t$ as follows. For every $w \in pos(t)$, we define

$$r(w) = \begin{cases} \underline{r}'(w) & \neg \exists 1 \leq j \leq m_{\kappa-1} : w_j^{\kappa-1} \leq_{pre} w \\ f & \text{if } \exists \kappa-1 \geq i \geq 1, 1 \leq j \leq m_i : w_i^j = w \\ \underline{r}_i^j(w) & \text{if } \exists \kappa-1 \geq i \geq 1, 1 \leq j \leq m_i : w_i^j <_{pre} w \\ & \text{and } \neg \exists 1 \leq j \leq m_{i-1} : w_j^{i-1} \leq_{pre} w \\ f & \text{if } lab(t,w) = \alpha \end{cases}$$

The calculation of the costs is the same as in the first of the proof of (*). This finishes the proof of Claim (*).

Now we can prove the correctness of the construction as follows (recall that $n = size_{\neg\alpha}(t)$):

$$\begin{aligned}
& (S_{M^{(*)}}, t) \\
= \ & \textstyle\sum_{p \in F^{(*)}} \sum_{r \in R_{M^{(*)}}(t,p)} \ c_{M^{(*)}}(t, r) \\
= \ & \textstyle\sum_{r \in R_{M^{(*)}}(t,f)} \ c_{M^{(*)}}(t, r) \\
= \ & \textstyle\sum_{x = (s, (u_j^i)_{\substack{n-1 \geq i \geq 1 \\ 1 \leq j \leq m_i}}) \in M_{\alpha,n}(t)} \sum_{\underline{r}' \in R_M(s, q_f)} \sum_{\underline{r}_j^i \in R_M(u_j^i, q_f)} \\
& \qquad c_M(s, \underline{r}') \odot \textstyle\prod_{\substack{n-1 \geq i \geq 1 \\ 1 \leq j \leq m_i}} c_M(u_j^i, \underline{r}_j^i) \\
& \text{(by Statement (*) using the fact that } A \text{ is idempotent)} \\
= \ & \textstyle\sum_{x = (s, (u_j^i)_{\substack{n-1 \geq i \geq 1 \\ 1 \leq j \leq m_i}}) \in M_{\alpha,n}(t)} \\
& \qquad \Big(\textstyle\sum_{\underline{r}' \in R_M(s, q_f)} c_M(s, \underline{r}')\Big) \odot \prod_{\substack{n-1 \geq i \geq 1 \\ 1 \leq j \leq m_i}} \Big(\sum_{\underline{r}_j^i \in R_M(u_j^i, q_f)} c_M(u_j^i, \underline{r}_j^i)\Big) \\
& \text{(by distributivity of } A) \\
= \ & \textstyle\sum_{x = (s, (u_j^i)_{\substack{n-1 \geq i \geq 1 \\ 1 \leq j \leq r_i}}) \in M_{\alpha,n}(t)} (S_M, s) \odot \prod_{\substack{n-1 \geq i \geq 1 \\ 1 \leq j \leq m_i}} (S_M, u_j^i) \\
& \text{(by definition of } S_M \text{; note that } q_f \text{ is the only final state of } M) \\
= \ & \textstyle\sum_{x \in M_{\alpha,n}(t)} \ c_{S_M}(x) \\
& \text{(by Definition 2.23)} \\
= \ & ((S_M)_\alpha^n, t) \\
& \text{(by Lemma 2.24)} \\
= \ & ((S_M)_\alpha^*, t) \\
& \text{(by Lemma 2.17(3))}
\end{aligned}$$

□

From Corollary 5.5 and Lemmas 5.2, 5.3, 5.4, 5.6, and 5.9 the following theorem is immediate.

**Theorem 5.10** Let $A$ be a commutative and idempotent semiring. Then

1. $A^{rec}\langle\langle T_\Sigma \rangle\rangle$ contains all polynomials tree series over $\Sigma$ and $A$ and
2. $A^{rec}\langle\langle T_\Sigma \rangle\rangle$ is closed under the rational operations.
3. $A^{rat}\langle\langle T_\Sigma \rangle\rangle \subseteq A^{rec}\langle\langle T_\Sigma \rangle\rangle$.

# 6 Kleene's result, comparison, and further research

Here we compare our result with other results from the literature and we indicate directions for furhter research. In this paper we have proved Kleene's result for formal tree series over commutative and idempotent semirings by showing automata-theoretic constructions with full correctness proofs (cf. Theorem 4.1 and Theorem 5.10).

Let $A$ be a commutative and idempotent semiring. Then
$A^{rec}\langle\langle T_\Sigma \rangle\rangle \subseteq \bigcup_{Q \text{ finite set}} A^{rat}\langle\langle T_\Sigma(Q) \rangle\rangle$ and
$A^{rat}\langle\langle T_\Sigma \rangle\rangle \subseteq A^{rec}\langle\langle T_\Sigma \rangle\rangle$.

Also in [BR82] and [Kui97a] Kleene's result has been extended to formal tree series. These approaches differ from our one in the requirements on the semiring $A$. Berstel and Reutenauer use fields. Since Kuich uses fixpoints of equations, he has to require that $A$ is continuous.

| | [BR82] | [Kui97a] | present paper |
|---|---|---|---|
| requirements on $A$ | field | commutative, naturally ordered, complete, and continuous | commutative and idempotent |

The definitions of these notions are as follows:

- A *field* is a commutative semiring $(A, \oplus, \odot, 0, 1)$ such that both $(A, \oplus, 0)$ and $(A - \{0\}, \odot, 1)$ are groups.
- $A$ is *naturally ordered* if the binary relation $\sqsubseteq$ on $A$ is a partial order on $A$, where $\sqsubseteq$ is defined by $a \sqsubseteq b$ iff there is a $c \in A$ such that $a \oplus c = b$,
- If $A$ is naturally ordered, then $A$ is *complete* if it is possible to define sums for all families $(a_i \,|\, i \in I)$ of elements of $A$, where $I$ is an index set, such that the following three conditions are satisfied:
  (i) $\sum_{i\in\emptyset} a_i = 0$, $\sum_{i\in\{j\}} a_i = a_j$, $\sum_{i\in\{j,k\}} a_i = a_j \oplus a_k$ for $j \neq k$
  (ii) $\sum_{j\in J}(\sum_{i\in I_j} a_i) = \sum_{i\in I} a_i$, if $\bigcup_{j\in J} I_j = I$ and $I_l \cap I_k = \emptyset$ for $l \neq k$
  (iii) $\sum_{i\in I}(c \odot a_i) = c \odot (\sum_{i\in I} a_i)$, $\sum_{i\in I}(a_i \odot c) = (\sum_{i\in I} a_i) \odot c$
- $A$ is *continuous* if for every sequence $(a_i \,|\, i \in \mathbb{N})$ in $A$, $sup\{\sum_{i\in E} a_i \,|\, E \subseteq I, E \text{ is finite}\} = \sum_{i\in I} a_i$.

Now let us compare these results. Let $\mathcal{C}_1$ be the class of fields, $\mathcal{C}_2$ be the class of semirings which are commutative, naturally ordered, complete, and continuous, and let $\mathcal{C}_3$ be the class of semirings which are commutative and idempotent. Obviously, $\mathcal{C}_1 \cap \mathcal{C}_2 = \mathcal{C}_1 \cap \mathcal{C}_3 = \emptyset$. The two classes $\mathcal{C}_2$ and $\mathcal{C}_3$ are incomparable:

- e.g. $\mathbb{B} \in \mathcal{C}_2 \cap \mathcal{C}_3$, also $Trop \in \mathcal{C}_2 \cap \mathcal{C}_3$ where $Trop$ is the tropical semiring defined by $Trop = (\mathbb{N} \cup \{\infty\}, min, +, \infty, 0)$ and the operations $min$ and $+$ are extended to $\mathbb{N} \cup \{\infty\}$ in the obvious way

- e.g. $Nat_\infty = (\mathbb{N} \cup \{\infty\}, +, \cdot, 0, 1) \in \mathcal{C}_2 - \mathcal{C}_3$ because $Nat_\infty$ is not idempotent
- any Boolean algebra belongs to $\mathcal{C}_3$, and to $\mathcal{C}_2$ iff it is a complete Boolean algebra
- e.g. $\mathbb{Q}_+ = (\mathbb{Q} \cup \{\infty\}, max, +, -\infty, 0) \in \mathcal{C}_3 - \mathcal{C}_2$ because $\mathbb{Q}_+$ is not continuous
- e.g. $A = ([0,1]_\mathbb{Q}, max, \cdot, 0, 1, )$ is idempotent but not continuous.

If $S$ is any submonoid of the monoid $([0,1], \cdot)$ with $0 \in S$, then the semiring $(S, max, \cdot, 0, 1)$ is idempotent and commutative. We might take $S = [0,1] \cap \mathbb{Q}$, and there are many further examples known from algebra which are dense in $[0,1]$, e.g., for any proper subset $T$ of the set of all primes, $S = \{\frac{a}{b} \in [0,1] \cap \mathbb{Q} \,|\, \forall p \in T : p \text{ does not divide } b\}$.

As open problem it remains to find a joint generalization of these three results, e.g., to the class of all commutative semirings.

## References


[Bak79] B. S. Baker. Composition of top-down and bottom-up tree transductions. *Inform. and Control*, 41:186–213, 1979.

[BGW00] A. L. Buchsbaum, R. Giancarlo, and J. R. Westbrook. On the determinization of weighted finite automata. *SIAM Journal on Computing*, 30(5):1502–1531, 2000.

[BR82] J. Berstel and C. Reutenauer. Recognizable formal power series in trees. *Theoretical Computer Science*, 18:115–148, 1982.

[BR88] J. Berstel and Ch. Reutenauer. *Rational Series and Their Languages*, volume 12 of *EATCS-Monographs*. Springer Verlag, 1988.

[CG95] R. A. Cuninghame-Green. Minimax algebra and applications. *Advances in Imaging and Electron Physics*, 90:1–121, 1995.

[CK93] K. Culik and J. Kari. Image compression using weighted finite automata. *Computer and Graphics*, 17:305–313, 1993.

[DG99] M. Droste and P. Gastin. The Kleene-Schützenberger theorem for formal power series in partially commuting variables. *Information and Computation*, 153:47–80, 1999. extended abstract in: *24th ICALP*, LNCS vol. 1256, Springer, 1997, pp. 682-692.

[DG00] M. Droste and P. Gastin. On aperiodic and star-free formal power series in partially commuting variables. In *Formal Power Series and Algebraic Combinatorics (Moscow 2000)*, pages 158–169. Springer, Berlin, 2000.

[EFV02] J. Engelfriet, Z. Fülöp, and H. Vogler. Bottom-up and top-down tree series transformations. *J. Automata, Languages and Combinatorics*, 2002. accepted.

[Eng75a] J. Engelfriet. Bottom–up and top–down tree transformations — a comparison. *Math. Systems Theory*, 9(3):198–231, 1975.

[Eng75b] J. Engelfriet. Tree automata and tree grammars. Technical Report DAIMI FN-10, Aarhus University, 1975.

[ES77] J. Engelfriet and E. M. Schmidt. IO and OI.I. *J. Comput. System Sci.*, 15(3):328–353, 1977.

[FSW94] C. Ferdinand, H. Seidl, and R. Wilhelm. Tree automata for code selection. *Acta Informatica*, 31:741–760, 1994.

[Gau01] S. Gaubert. Workshop on max-plus-algebras and their applications to discrete-event systems, theoretical computer science, and optimization. Prague, Czech Republic, August 2001. International Federation of Automatic Control (IFAC), Prague.

[GP97] S. Gaubert and M. Plus. Methods and applications of (max,+) linear algebra. Technical Report 3088, INRIA, Rocquencourt, January 1997.

[GS84] F. Gécseg and M. Steinby. *Tree Automata*. Akadémiai Kiadó, Budapest, 1984.

[GS97] F. Gécseg and M. Steinby. Tree languages. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 3, chapter 1, pages 1–68. Springer-Verlag, 1997.

[Haf99] U. Hafner. *Low Bit-Rate Image and Video Coding with Weighted Finite Automata*. PhD thesis, Universität Würzburg, Germany, 1999.

[JLdV00] Z. Jiang, B. Litow, and O. de Vel. Similarity enrichment in image compression through weighted finite automata. In *COCOON 2000*, LNCS 1858, pages 447–456. Springer, 2000.

[Kat01] F. Katritzke. *Refinements of data compression using weighted finite automata*. PhD thesis, Siegen University, 2001.

[Kle56] S. E. Kleene. Representation of events in nerve nets and finite automata. In C.E. Shannon and J. McCarthy, editors, *Automata Studies*, pages 3–42. Princeton University Press, Princeton, N.J., 1956.

[KS86] W. Kuich and A. Salomaa. *Semirings, Automata, Languages*. EATCS Monographs on Theoretical Computer Science, Springer Verlag, 1986.

[Kui97a] W. Kuich. Formal power series over trees. In *Proc. of the 3rd International Conference Developments in Language Theory*, pages 60–101. Aristotle University of Thessaloniki, 1997.

[Kui97b] W. Kuich. Semirings and formal power series: Their relevance to formal languages and automata. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages, Vol. 1*, chapter 9, pages 609–677. Springer, 1997.

[Kui98] W. Kuich. Tree transducers and formal tree series. 1998.

[Kui99] W. Kuich. Linear systems of equations and automata on distributive multioperator monoids. In *Constributions to General Algebra*, number 12, pages 1–10. Proceedings of the Vienna Conference, Verlag Johannes Heyn, June 1999.

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

[MPR00] M. Mohri, F. Pereira, and M. Riley. The design principles of a weighted finite-state transducer library. *Theoret. Comp. Science*, 231:17–32, 2000.

[Per90] D. Perrin. Finite automata. In J. van Leuuwen, editor, *Handbook of Theoretical Computer Science, Volume B*, chapter 1, pages 3–57. Elsevier Science Publisher B.V., 1990.

[Rou70] W.C. Rounds. Mappings and grammars on trees. *Math. Systems Theory*, 4:257–287, 1970.

[Sch61] M.P. Schützenberger. On the definition of a family of automata. *Information and Control*, 4:245–270, 1961.

[Sei92] H. Seidl. Finite tree automata with cost functions. In *17th CAAP*, volume 581 of *Lect. Notes Comput. Sci.*, pages 279–299. Springer–Verlag, 1992.

[Sei94] H. Seidl. Finite tree automata with cost functions. *Theoret. Comput. Sci.*, 126:113–142, 1994.

[SS78] A. Salomaa and M. Soittola. *Automata-Theoretic Aspects of Formal Power Series*. Texts and Monographs in Computer Science, Springer Verlag, 1978.

[Tha70] J.W. Thatcher. Generalized$^2$ sequential machine maps. *J. Comput. System Sci.*, 4:339–367, 1970.

[TW68] J.W. Thatcher and J.B. Wright. Generalized finite automata theory with application to a decision problem of second-order logic. *Math. Systems Theory*, 2(1):57–81, 1968.

[Zim81] U. Zimmermann. *Linear and Combinatorial Optimization in Ordered Algebraic Structures*. North-Holland, Amsterdam - New York - Oxford, 1981.