Relating Tree Series Transducers and Weighted Tree Automata

Andreas Maletti*

Technische Universität Dresden
Fakultät Informatik
D–01062 Dresden, Germany.
maletti@tcs.inf.tu-dresden.de

Abstract. In this paper we implement bottom-up tree series transducers (tst) over the semiring $\mathcal{A}$ with the help of bottom-up weighted tree automata (wta) over an extension of $\mathcal{A}$. Therefore we firstly introduce bottom-up DM-monoid weighted tree automata (DM-wta), which essentially are wta using an operation symbol of a DM-monoid instead of a semiring element as transition weight. Secondly, we show that DM-wta are indeed a generalization of tst (using pure substitution). Thirdly, given a DM-wta over a DM-monoid we construct a semiring $\mathcal{A}$ along with a wta such that the wta computes a formal representation of the semantics of the DM-wta.
Finally, we demonstrate the applicability of our presentation result by deriving a pumping lemma for deterministic tst as well as deterministic DM-wta from a pumping lemma for deterministic wta.

1 Introduction

In formal language theory several different accepting and transducing devices were intensively studied [1]. A classical folklore result shows how to implement generalized sequential machines (cf., e.g., [2]) on weighted automata [3–5] with the help of the particular semiring $(\mathfrak{P}(\Sigma^*), \cup, \circ)$ of languages over the alphabet Σ. Naturally, this semiring is not commutative, notwithstanding the representation allows us to transfer results obtained for weighted automata to generalized sequential machines. In this sense, the study of arbitrary weighted automata subsumes the study of generalized sequential machines.

We translate the above representation result to tree languages (cf., e.g., [6]), i.e., we show how to implement bottom-up tree transducers [7, 8] on bottom-up weighted tree automata (wta) [9, 10]. More generally, we even unearth a relationship between bottom-up tree series transducers (tst) [11, 12] using pure substitution and wta. Therefore we first introduce bottom-up DM-monoid weighted tree automata (DM-wta), which essentially are wta where the weight of a transition is an operation symbol of a DM-monoid [10] instead of a semiring element. These devices can easily simulate both wta and tst by a proper choice of the DM-monoid (cf. Proposition 5). Next we devise a monoid $\mathcal{A}$ which is capable of em-

* Financially supported by the German Research Foundation (DFG, GK 334/3)

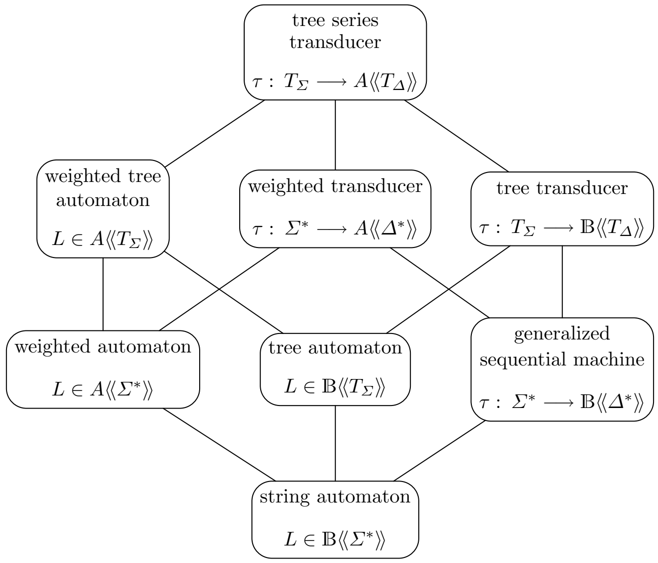

Fig. 1. Generalization hierarchy

ulating the effect of the operation symbols of a DM-monoid $\mathcal{D}$ (cf. Theorem 6). Then we extend $\mathcal{A}$ to a semiring using the addition of a semiring $\mathcal{B}$ for which $\mathcal{D}$ is a semimodule (cf. Theorem 8). In this way we obtain an abstract addition (of $\mathcal{B}$), which allows us to perform the concrete addition (of $\mathcal{D}$) later. Thereby we obtain a representation result, in which a tst or a DM-wta is presented as wta, which computes a formal representation of the semantics of the tst or DM-wta.

For a tst M over a completely idempotent semiring $\mathcal{A}$, e.g., all tree transducers, we can refine the constructed semiring with the help of a congruence relation such that the factor semiring uses (an extension of) the concrete addition of $\mathcal{A}$ (cf. Theorem 10). Then one can construct a wta such that it computes the same tree series as M. Finally, we note that the construction of the semiring preserves many beneficial properties (concerning the addition) of the original DM-monoid.

Hence the study of wta subsumes the study of tst over completely idempotent semirings. In fact, the subsumption also holds for deterministic devices, i.e., the study of deterministic wta subsumes the study of deterministic tst or DM-wta. To illustrate the applicability of the relationship we transfer a pumping lemma [13] for deterministic finite wta to both tst and DM-wta. This is possible, because the semiring addition is irrelevant for deterministic wta and the determinism property is preserved by the constructions. This yields that for a given tst or DM-wta M we can construct a wta M' such that $\|M'\| = \|M\|$. Hence the pumping lemma for wta can readily be transfered to tst and DM-wta.

2 Preliminaries

The set $\{0, 1, 2, \ldots\}$ of all non-negative integers is denoted by $\mathbb{N}$ and we let $\mathbb{N}_+ = \mathbb{N} \setminus \{0\}$. In the following let $k, n \in \mathbb{N}$. The interval $[k, n]$ abbreviates $\{ i \in \mathbb{N} \mid k \leq i \leq n \}$ and we use $[n]$ to stand for $[1, n]$. The set of all subsets of a set A is denoted by $\mathfrak{P}(A)$ and the set of all (total) mappings $f \colon A \longrightarrow B$ is denoted by B^A as customary. Finally, the set of all words over A is displayed as A^*, the length of a word $w \in A^*$ is denoted by $|w|$, and $\cdot$ is used to denote concatenation as well as to delimit subwords.

2.1 Trees and Substitutions

A non-empty set Σ equipped with a mapping $\mathrm{rk}_\Sigma \colon \Sigma \longrightarrow \mathbb{N}$ is called an *operator alphabet*. The set $\Sigma_k = \{ \sigma \in \Sigma \mid \mathrm{rk}_\Sigma(\sigma) = k \}$ denotes the set of operators of arity k. Given a set V, the set $T_\Sigma(V)$ of (finite, labeled, and ordered) Σ-trees indexed by V is the smallest set T such that $\Sigma_0 \cup V \subseteq T$ and for every $k \in \mathbb{N}$, $\sigma \in \Sigma_k$, and $t_1, \ldots, t_k \in T$ also $\sigma(t_1, \ldots, t_k) \in T$. The notation T_Σ abbreviates $T_\Sigma(\emptyset)$. The mapping $\mathrm{pos} \colon T_\Sigma(V) \longrightarrow \mathfrak{P}(\mathbb{N}^*)$ is defined for every $v \in V$, $\sigma \in \Sigma_k$, and $t_1, \ldots, t_k \in T_\Sigma(V)$ by

$$\mathrm{pos}(v) = \{\varepsilon\} \qquad \mathrm{pos}(\sigma(t_1, \ldots, t_k)) = \{\varepsilon\} \cup \{ i \cdot w_i \mid i \in [k], w_i \in \mathrm{pos}(t_i) \} \enspace .$$

Moreover, $\mathrm{height}(t) = 1 + \max\{ |w| \mid w \in \mathrm{pos}(t) \}$ for every $t \in T_\Sigma(V)$. The *label* of t at $w \in \mathrm{pos}(t)$ is denoted by $\mathrm{lab}_t(w)$, i.e.,

$$\mathrm{lab}_v(\varepsilon) = v \qquad \mathrm{lab}_{\sigma(t_1, \ldots, t_k)}(w) = \begin{cases} \sigma & \text{, if } w = \varepsilon \\ \mathrm{lab}_{t_i}(w_i) & \text{, if } w = i \cdot w_i \text{ with } i \in [k] \end{cases} \enspace .$$

For convenience, we assume a countably infinite set $X = \{ x_i \mid i \in \mathbb{N}_+ \}$ of formal variables and its finite subsets $X_n = \{ x_i \mid i \in [n] \}$. A Σ-tree $t \in T_\Sigma(X_n)$ is in the set $\widehat{T_\Sigma}(X_n)$, if and only if every $x \in X_n$ occurs exactly once in t. Given $t \in T_\Sigma(X_n)$ and $t'_1, \ldots, t'_n \in T_\Sigma(V)$, the expression $t[t'_1, \ldots, t'_n]$ denotes the *(parallel) tree substitution* of t'_i for every occurrence of x_i in t, i.e., $x_i[t'_1, \ldots, t'_n] = t'_i$ for every $i \in [n]$ and

$$\sigma(t_1, \ldots, t_k)[t'_1, \ldots, t'_n] = \sigma(t_1[t'_1, \ldots, t'_n], \ldots, t_k[t'_1, \ldots, t'_n])$$

for every $k \in \mathbb{N}$, $\sigma \in \Sigma_k$, and $t_1, \ldots, t_k \in T_\Sigma(X_n)$. Let $t \in \widehat{T_\Sigma}(X_n)$ with $n \geq 1$ and $t' \in \widehat{T_\Sigma}(X_k)$. The *non-identifying tree substitution* of t' into t, denoted by $t\langle\!\langle t' \rangle\!\rangle$, yields a tree of $\widehat{T_\Sigma}(X_{k+n-1})$ which is defined by

$$t\langle\!\langle t' \rangle\!\rangle = t[t', x_{k+1}, \ldots, x_{k+n-1}] \enspace .$$

This way no variable of t' is identified with a variable of t. To complete the definition we let $t\langle\!\langle t' \rangle\!\rangle = t$ whenever $t \in T_\Sigma$, i.e., $n = 0$. One can compare this with the classical lambda-calculus, where (except for reordering of the arguments)

$$(\lambda x_1 \ldots x_n.t)(\lambda x_1 \ldots x_k.t') \Rightarrow \lambda x_1 \ldots x_{k+n-1}.t\langle\!\langle t' \rangle\!\rangle \enspace .$$

2.2 Algebraic structures

Given a carrier set A, an operator alphabet Ω, and a family $I = (I_k)_{k\in\mathbb{N}}$ of mappings $I_k : \Omega_k \longrightarrow A^{(A^k)}$ interpreting the symbols as operations on A, the triple (A, Ω, I) is called an *(abstract) Ω-algebra*. The algebra (T_Ω, Ω, I) where $I_k(\omega) = \overline{\omega}$ for every $k \in \mathbb{N}$, $\omega \in \Omega_k$, and $\overline{\omega}(t_1, \ldots, t_k) = \omega(t_1, \ldots, t_k)$ for every $t_1, \ldots, t_k \in T_\Omega$ is called the *initial (term) Ω-algebra*. In the sequel we often do not differentiate between the symbol and the actual operation. Usually the context will provide sufficient information as to clarify which meaning is intended. Further we occasionally omit the operator alphabet and instead list the operators and identify nullary operators with elements of A.

Monoids are algebraic structures $\mathcal{A} = (A, \otimes)$ with carrier set A, an *associative* operation $\otimes : A^2 \longrightarrow A$, i.e., $a_1 \otimes (a_2 \otimes a_3) = (a_1 \otimes a_2) \otimes a_3$ for every $a_1, a_2, a_3 \in A$, and a *neutral element* $1 \in A$, i.e., $1 \otimes a = a = a \otimes 1$ for every $a \in A$. The neutral element is unique and denoted by $0_\mathcal{A}$ or $1_\mathcal{A}$ in the sequel. The monoid is said to be *commutative*, if $a_1 \otimes a_2 = a_2 \otimes a_1$ for every $a_1, a_2 \in A$, and it is said to be *idempotent*, if $a = a \otimes a$ for every $a \in A$. A commutative monoid is called *complete*, if it is possible to define an (infinitary) operation $\bigotimes$ such that the following two additional axioms hold for all index sets I, J and all families $(a_i)_{i\in I}$ of monoid elements.

(i) $\bigotimes_{i\in\{j\}} a_i = a_j$ and $\bigotimes_{i\in\{j_1,j_2\}} a_i = a_{j_1} \otimes a_{j_2}$ for $j_1 \neq j_2$.
(ii) $\bigotimes_{j\in J} \bigotimes_{i\in I_j} a_i = \bigotimes_{i\in I} a_i$, if $\bigcup_{j\in J} I_j = I$ and $I_{j_1} \cap I_{j_2} = \emptyset$ for $j_1 \neq j_2$.

The relation $\sqsubseteq\, \subseteq A^2$ is defined by $a_1 \sqsubseteq a_2$ if and only if there exists $a \in A$ such that $a_1 \otimes a = a_2$. If $\sqsubseteq$ is a partial order, then $\mathcal{A}$ is said to be *naturally ordered*. Finally, a naturally ordered and complete monoid is *continuous*, if for every $a \in A$, index set I, and family $(a_i)_{i\in I}$ of elements $a_i \in A$

$$\bigotimes_{i\in E} a_i \sqsubseteq a \text{ for all finite } E \subseteq I \quad\iff\quad \bigotimes_{i\in I} a_i \sqsubseteq a \ .$$

Note that an idempotent monoid is continuous, if and only if it is completely idempotent, i.e., it is complete and for every non-empty index set I and element $a \in A$ we have that $\bigotimes_{i\in I} a = a$.

Algebraic structures $\mathcal{A} = (A, \oplus, \odot)$ made of two monoids $(A, \oplus)$ and $(A, \odot)$ with neutral elements $0_\mathcal{A}$ and $1_\mathcal{A}$, respectively, of which the former monoid is commutative and the latter monoid has $0_\mathcal{A}$ as an absorbing element, i.e., $a \odot 0_\mathcal{A} = 0_\mathcal{A} = 0_\mathcal{A} \odot a$ for every $a \in A$, are called *semirings (with one and absorbing zero)*, if the monoids are connected via the distributivity laws, i.e., $a_1 \odot (a_2 \oplus a_3) = (a_1 \odot a_2) \oplus (a_1 \odot a_3)$ and $(a_1 \oplus a_2) \odot a_3 = (a_1 \odot a_3) \oplus (a_2 \odot a_3)$ for every $a_1, a_2, a_3 \in A$. The semiring $\mathcal{A}$ is called *(additively) idempotent*, if $(A, \oplus)$ is idempotent. Finally, a *complete* semiring consists of a complete monoid $(A, \oplus)$ and satisfies the additional constraint that for every index set I, $a \in A$, and family $(a_i)_{i\in I}$ of semiring elements

$$\bigoplus_{i\in I} (a \odot a_i) = a \odot \bigoplus_{i\in I} a_i \quad\text{and}\quad \bigoplus_{i\in I} (a_i \odot a) = \Big(\bigoplus_{i\in I} a_i\Big) \odot a \ .$$

Let $\mathcal{B} = (B, +)$ be a commutative monoid, $\mathcal{A} = (A, \oplus, \odot)$ be a semiring, and $\cdot : A \times B \longrightarrow B$ be a mapping. Then $\mathcal{B}$ is called a *(left) $\mathcal{A}$-semimodule (via $\cdot$)*, if the conditions (i)-(iii) hold for all $a, a_1, a_2 \in A$ and all $b, b_1, b_2 \in B$.

(i) $a \cdot 0_{\mathcal{B}} = 0_{\mathcal{B}}$ and $1_{\mathcal{A}} \cdot b = b$.
(ii) $(a_1 \odot a_2) \cdot b = a_1 \cdot (a_2 \cdot b)$.
(iii) $a \cdot (b_1 + b_2) = (a \cdot b_1) + (a \cdot b_2)$ and $(a_1 \oplus a_2) \cdot b = (a_1 \cdot b) + (a_2 \cdot b)$.

Given that $\mathcal{B}$ and $\mathcal{A}$ are complete, $\mathcal{B}$ is called a *complete $\mathcal{A}$-semimodule*, if for every family $(\, b_i \,)_{i\in I}$ of monoid elements and family $(\, a_i \,)_{i\in I}$ of semiring elements the additional axioms (iv) and (v) hold.

(iv) $a \cdot \sum_{i\in I} b_i = \sum_{i\in I} (a \cdot b_i)$
(v) $\left(\bigoplus_{i\in I} a_i\right) \cdot b = \sum_{i\in I} (a_i \cdot b)$

Clearly each commutative monoid $\mathcal{B} = (B, +)$ is an $\mathbb{N}$-semimodule, where the semiring of non-negative integers is given by $(\mathbb{N}, +, \cdot)$, using the mixed operation $\cdot : \mathbb{N} \times B \longrightarrow B$ defined as $n \cdot b = \sum_{i\in[n]} b$ for every $n \in \mathbb{N}$ and $b \in B$. Note that $\sum_{i\in[0]} b = 0_{\mathcal{B}}$. Similarly, every commutative and continuous monoid is a complete $\mathbb{N}_\infty$-semimodule (cf. [10]), where $\mathbb{N}_\infty = (\mathbb{N} \cup \{+\infty\}, +, \cdot)$. Furthermore, any idempotent and commutative monoid $\mathcal{B}$ is a $\mathbb{B}$-semimodule where $\mathbb{B} = (\{0, 1\}, \vee, \wedge)$ is the *boolean semiring*, and $\mathcal{B}$ is a complete $\mathbb{B}$-semimodule, if $\mathcal{B}$ additionally is completely idempotent (cf. [10]).

Let (D, Ω) be an Ω-algebra. The algebraic structure $\mathcal{D} = (D, +, \Omega)$ is called a *distributive multi-operator monoid* (DM-monoid) [10], if $(D, +)$ is a commutative monoid with neutral element $0_{\mathcal{D}}$ and for every $k \in \mathbb{N}$, $\omega \in \Omega_k$, $i \in [k]$, and $d, d_1, \ldots, d_k \in D$

(i) $\omega(d_1, \ldots, d_{i-1}, 0_{\mathcal{D}}, d_{i+1}, \ldots, d_k) = 0_{\mathcal{D}}$,
(ii) $\omega(d_1, \ldots, d_{i-1}, d + d_i, d_{i+1}, \ldots, d_k) = \omega(d_1, \ldots, d, \ldots, d_k) + \omega(d_1, \ldots, d_k)$.

For $\mathcal{D}$ to be *complete* we demand that $(D, +)$ is complete and for every $k \in \mathbb{N}$, $\omega \in \Omega_k$, index sets $I_1, \ldots, I_k$, and family $(\, d_i \,)_{i\in I_j}$ of monoid elements for every $j \in [k]$ the equality

$$\omega\Big(\sum_{i_1\in I_1} d_{i_1}, \ldots, \sum_{i_k\in I_k} d_{i_k}\Big) = \sum_{i_1\in I_1} \cdots \sum_{i_k\in I_k} \omega(d_{i_1}, \ldots, d_{i_k})$$

is satisfied. Finally, $\mathcal{D}$ is *continuous*, if $\mathcal{D}$ is complete and $(D, +)$ is continuous.

The DM-monoid $\mathcal{D}$ is said to be an *$\mathcal{A}$-semimodule* for some commutative semiring $\mathcal{A} = (A, \oplus, \odot)$, if $(D, +)$ is an $\mathcal{A}$-semimodule and for every $k \in \mathbb{N}$, $\omega \in \Omega_k$, $a \in A$, $i \in [k]$, and $d_1, \ldots, d_k \in D$ the equality

$$\omega(d_1, \ldots, d_{i-1}, a \cdot d_i, d_{i+1}, \ldots, d_k) = a \cdot \omega(d_1, \ldots, d_k)$$

holds. The DM-monoid $\mathcal{D}$ is a *complete $\mathcal{A}$-semimodule*, if both $\mathcal{A}$ and $\mathcal{D}$ are by itself complete and for every $a \in A$, $d \in D$, index set I, and family $(\, a_i \,)_{i\in I}$ and $(\, d_i \,)_{i\in I}$ of semiring and monoid elements, respectively, we have

$$\Big(\bigoplus_{i\in I} a_i\Big) \cdot d = \sum_{i\in I} (a_i \cdot d) \qquad \text{and} \qquad a \cdot \sum_{i\in I} d_i = \sum_{i\in I} (a \cdot d_i)\ .$$

Clearly, every DM-monoid is an $\mathbb{N}$-semimodule.

2.3 Formal Power Series and Tree Series Substitution

Any mapping $\varphi : T \longrightarrow A$ from a set T into a commutative monoid $\mathcal{A} = (A, \oplus)$ is also called *(formal) power series*. The set of all power series is denoted by $A\langle\langle T\rangle\rangle$. We write (φ, t) instead of $\varphi(t)$ for $\varphi \in A\langle\langle T\rangle\rangle$ and $t \in T$. The *sum* $\varphi_1 \oplus \varphi_2$ of two power series $\varphi_1, \varphi_2 \in A\langle\langle T\rangle\rangle$ is defined pointwise by $(\varphi_1 \oplus \varphi_2, t) = (\varphi_1, t) \oplus (\varphi_2, t)$ for every $t \in T$. The *support* $\mathrm{supp}(\varphi)$ of φ is defined by

$$\mathrm{supp}(\varphi) = \{ \, t \in T \mid (\varphi, t) \neq 0_{\mathcal{A}} \, \} \ .$$

If the support of φ is finite, then φ is said to be a *polynomial*. The power series with empty support is denoted by $\widetilde{0}_{\mathcal{A}}$.

In case $T = T_\Sigma(V)$ for some ranked alphabet Σ and set V, then φ is also called *(formal) tree series*. Let $\mathcal{A} = (A, \oplus, \odot)$ now be a complete semiring and let $n \in \mathbb{N}$, $\varphi \in A\langle\langle T_\Sigma(X_n)\rangle\rangle$, and $\psi_1, \ldots, \psi_n \in A\langle\langle T_\Sigma\rangle\rangle$. We define the *tree series substitution* of $(\psi_1, \ldots, \psi_n)$ into φ, denoted by $\varphi \longleftarrow (\psi_1, \ldots, \psi_n)$, as

$$\varphi \longleftarrow (\psi_1, \ldots, \psi_n) = \bigoplus_{\substack{t \in T_\Sigma(X_n),\\ t_1, \ldots, t_n \in T_\Sigma}} \Big((\varphi, t) \odot \bigodot_{i \in [n]} (\psi_i, t_i) \Big) \, t[t_1, \ldots, t_n] \ .$$

Note that the order in the product is given by the order $1 < \cdots < n$ of the indices. Furthermore, note that irrespective of the number of occurrences of x_i the coefficient (ψ_i, t_i) is taken into account exactly once, even if x_i does not appear at all in t. This notion of substitution is called *pure IO-substitution* [11]. Other notions of substitution, like o-IO-substitution [12] and OI-substitution [14], have been defined, but in this paper we will exclusively deal with pure IO-substitution.

2.4 Tree Automata and Tree Series Transducers

Let I and J be sets. An $(I \times J)$-*matrix* over a set S is a mapping $M : I \times J \longrightarrow S$. The set of all $(I \times J)$-matrices is denoted by $S^{I \times J}$ and the (i, j)-entry with $i \in I$ and $j \in J$ of a matrix $M \in S^{I \times J}$ is usually denoted by $M_{i,j}$ instead of $M(i, j)$. Let Σ be an operator alphabet, I be a non-empty set, and $\mathcal{A} = (A, \oplus)$ be a commutative monoid. Every family $\mu = (\, \mu_k \,)_{k \in \mathbb{N}}$ of mappings $\mu_k : \Sigma_k \longrightarrow A^{I \times I^k}$ is called *tree representation* over Σ, I, and A. A *deterministic* tree representation additionally fulfills the restriction that for every $\sigma \in \Sigma_k$ and $i_1, \ldots, i_k \in I$ there exists at most one $i \in I$ such that $\mu_k(\sigma)_{i,(i_1,\ldots,i_k)} \neq 0_{\mathcal{A}}$.

A *(bottom-up) weighted tree automaton* (wta) is a system $M = (I, \Sigma, \mathcal{A}, F, \mu)$ comprising of a set I of *states*, a finite input ranked alphabet Σ, a semiring $\mathcal{A} = (A, \oplus, \odot)$, a vector $F \in A^I$ of *final weights*, and a tree representation μ over Σ, I, and A. If I is infinite, then $\mathcal{A}$ must be complete, otherwise M is called *finite*. Moreover, M is *deterministic*, if μ is deterministic. Let $\boldsymbol{\mu} = (\, \boldsymbol{\mu}_k(\sigma) \,)_{k \in \mathbb{N}, \sigma \in \Sigma_k}$ where $\boldsymbol{\mu}_k(\sigma) : (A^I)^k \longrightarrow A^I$ is defined componentwise for every $i \in I$ and $V_1, \ldots, V_k \in A^I$ by

$$\boldsymbol{\mu}_k(\sigma)(V_1, \ldots, V_k)_i = \bigoplus_{i_1, \ldots, i_k \in I} \mu_k(\sigma)_{i,(i_1,\ldots,i_k)} \odot (V_1)_{i_1} \odot \cdots \odot (V_k)_{i_k} \ .$$

Let $h_\mu : T_\Sigma \longrightarrow A^I$ be the unique homomorphism from (T_Σ, Σ) to $(A^I, \boldsymbol{\mu})$. The tree series $\|M\| \in A\langle\langle T_\Sigma \rangle\rangle$ recognized by M is $(\|M\|, t) = \bigoplus_{i \in I} F_i \odot h_\mu(t)_i$ for every $t \in T_\Sigma$.

A *(bottom-up) tree series transducer* (tst) M is a system $(I, \Sigma, \Delta, \mathcal{A}, F, \mu)$ in which I is a set of *states*, Σ and Δ are finite input and output ranked alphabets, respectively, $\mathcal{A} = (A, \oplus, \odot)$ is a semiring, $F \in A\langle\langle T_\Delta(X_1) \rangle\rangle^I$ is a vector of *final outputs*, and μ is a tree representation over Σ, I, and $A\langle\langle T_\Delta(X) \rangle\rangle$ such that $\mu_k(\sigma) \in A\langle\langle T_\Delta(X_k) \rangle\rangle^{I \times I^k}$ for every $k \in \mathbb{N}$ and $\sigma \in \Sigma_k$. If I is finite and each tree series in the range of $\mu_k(\sigma)$ is a polynomial, then M is called *finite*, otherwise $\mathcal{A}$ must be complete. Finite tst over the Boolean semiring $\mathbb{B}$ are also called *tree transducer*. The tst M is *deterministic*, if μ is deterministic. Let $\boldsymbol{\mu} = (\, \boldsymbol{\mu}_k(\sigma) \,)_{k \in \mathbb{N}, \sigma \in \Sigma_k}$ where $\boldsymbol{\mu}_k(\sigma) : (A\langle\langle T_\Delta \rangle\rangle^I)^k \longrightarrow A\langle\langle T_\Delta \rangle\rangle^I$ is defined componentwise for every $i \in I$ and $V_1, \ldots, V_k \in A\langle\langle T_\Delta \rangle\rangle^I$ by

$$\boldsymbol{\mu}_k(\sigma)(V_1, \ldots, V_k)_i = \bigoplus_{i_1, \ldots, i_k \in I} \mu_k(\sigma)_{i,(i_1,\ldots,i_k)} \longleftarrow ((V_1)_{i_1}, \ldots, (V_k)_{i_k}) \enspace .$$

Let $h_\mu : T_\Sigma \longrightarrow A\langle\langle T_\Delta \rangle\rangle^I$ be the unique homomorphism from the initial Σ-algebra (T_Σ, Σ) to $(A\langle\langle T_\Delta \rangle\rangle^I, \boldsymbol{\mu})$. For every $t \in T_\Sigma$ the *tree-to-tree-series transformation* (t-ts transformation) $\|M\| : T_\Sigma \longrightarrow A\langle\langle T_\Delta \rangle\rangle$ computed by M is $(\|M\|, t) = \bigoplus_{i \in I} F_i \longleftarrow (h_\mu(t)_i)$.

3 Establishing the relationship

Inspired by the automaton definition of [10] we define DM-monoid weighted tree automata (DM-wta). Roughly speaking, to each transition of a DM-wta an operation symbol of a DM-monoid is associated.

Definition 1. *A* DM-monoid weighted tree automaton *(DM-wta) is a system $M = (I, \Sigma, \mathcal{D}, F, \mu)$, where*

- *I is a non-empty set of* states*,*
- *Σ is a finite operator alphabet of* input symbols*,*
- *$\mathcal{D} = (D, +, \Omega)$ is a DM-monoid,*
- *$F \in (\Omega_1)^I$ is the* final weight vector*, and*
- *$\mu = (\, \mu_k \,)_{k \in \mathbb{N}}$ is a tree representation over I, Σ, and Ω.*

If I is infinite, then $\mathcal{D}$ must be complete. Otherwise, M is called finite*. Finally, M is* deterministic*, if μ is deterministic.*

Unless stated otherwise let $M = (I, \Sigma, \mathcal{D}, F, \mu)$ be a DM-wta over the DM-monoid $\mathcal{D} = (D, +, \Omega)$. In the following let $k \in \mathbb{N}$, $\sigma \in \Sigma_k$, $i \in I$, and $t = \sigma(t_1, \ldots, t_k) \in T_\Sigma$. Moreover, all function arguments range over their respective domains. Next we define two semantics, namely initial algebra semantics [15] and a semantics based on runs. In the latter the weight of a run is obtained by combining the weights obtained for the direct subtrees with the help of the operation symbol associated to the topmost transition. Nondeterminism is taken care of by adding the weights of all runs on a given input tree.

Definition 2. *Let* $\boldsymbol{\mu} = (\,\boldsymbol{\mu}_k(\sigma)\,)_{k\in\mathbb{N},\sigma\in\Sigma_k}$ *where* $\boldsymbol{\mu}_k(\sigma) : (D^I)^k \longrightarrow D^I$ *is defined componentwise for every* $i \in I$ *by*

$$\boldsymbol{\mu}_k(\sigma)(V_1,\ldots,V_k)_i = \sum_{i_1,\ldots,i_k\in I} \mu_k(\sigma)_{i,(i_1,\ldots,i_k)}((V_1)_{i_1},\ldots,(V_k)_{i_k}) \enspace .$$

Let $h_\mu : T_\Sigma \longrightarrow D^I$ *be the unique homomorphism from* (T_Σ,Σ) *to* $(D^I,\boldsymbol{\mu})$*. The* tree series recognized by M *is defined as* $(\|M\|,t) = \sum_{i\in I} F_i(h_\mu(t)_i)$.

Definition 3. *A* run on $t \in T_\Sigma$ *is a mapping* $r : \mathrm{pos}(t) \longrightarrow I$*. The set of all runs on* t *is denoted by* $R(t)$*. The* weight of r *is defined by the mapping* $\mathrm{wt}_r : \mathrm{pos}(t) \longrightarrow D$ *which is defined for* $w \in \mathrm{pos}(t)$ *with* $\mathrm{lab}_t(w) \in \Sigma_k$ *by*

$$\mathrm{wt}_r(w) = \mu_k(\mathrm{lab}_t(w))_{r(w),(r(w\cdot 1),\ldots,r(w\cdot k))}(\mathrm{wt}_r(w\cdot 1),\ldots,\mathrm{wt}_r(w\cdot k)) \enspace .$$

The run-based semantics of M *is* $(|M|,t) = \sum_{r\in R(t)} F_{r(\varepsilon)}(\mathrm{wt}_r(\varepsilon))$.

The next proposition states that the initial algebra semantics coincides with the run-based semantics, which is mainly due to the distributivity of the DM-monoid. Intuitively speaking, this reflects the property that nondeterminism can equivalently either be handled locally (initial algebra semantics) or globally (run-based semantics).

Proposition 4. *For every DM-wta* M *we have* $\|M\| = |M|$.

The next proposition demonstrates how powerful DM-wta are. In fact, every wta and every tst can be simulated by a DM-wta.

Proposition 5. *Let* M_1 *be a wta and* M_2 *be a tst.*

(i) There exists a DM-wta M *such that* $\|M\| = \|M_1\|$.
(ii) There exists a DM-wta M *such that* $\|M\| = \|M_2\|$.

Proof. Since it is clear (cf. [11]), how to simulate a wta with the help of a tst, we only show Statement (ii). Let $M_2 = (I_2,\Sigma,\Delta,\mathcal{A},F_2,\mu_2)$ *be a tst,*

$$\Omega = \{\, \underline{\varphi}_k \mid k \in \mathbb{N}, \varphi \in A\langle\langle T_\Delta(X_k)\rangle\rangle \,\} \enspace ,$$

and let $\underline{\varphi}_k : A\langle\langle T_\Delta\rangle\rangle^k \longrightarrow A\langle\langle T_\Delta\rangle\rangle$ *be defined as*

$$\underline{\varphi}_k(\psi_1,\ldots,\psi_k) = \varphi \longleftarrow (\psi_1,\ldots,\psi_k) \enspace .$$

Then, by [10, 11], $\mathcal{D} = (A\langle\langle T_\Delta\rangle\rangle,\oplus,\Omega)$ *is a DM-monoid, which is complete whenever* $\mathcal{A}$ *is. Hence we let* $M = (I_2,\Sigma,\mathcal{D},F,\mu)$ *with* $F_i = \underline{F_2(i)}_1$ *and for every* $i,i_1,\ldots,i_k \in I_2$ *we set* $\mu_k(\sigma)_{i,(i_1,\ldots,i_k)} = \underline{(\mu_2)_k(\sigma)_{i,(i_1,\ldots,i_k)}}_k$.

Note that in both statements of Proposition 5, M can be constructed to be deterministic, whenever the input device, i.e., M_1 or M_2, is deterministic. Let $\mathcal{D} = (D,\Omega)$ be an Ω-algebra. In the following ω ranges over Ω_k. We denote by ΩX the set of all terms $\{\,\overline{\omega}(x_1,\ldots,x_k) \mid \omega \in \Omega_k\,\}$. We can define a monoid which simulates the algebra $\mathcal{D}$ as follows. Recall that we use overlining, if we want to refer to the term obtained by top-concatenation of the overlined symbol with its arguments.

Theorem 6. *For every Ω-algebra (D, Ω) there exists a monoid $(B, \leftarrow)$ such that $D \cup \Omega X \subseteq B$ and for all $d_1, \ldots, d_k \in D$*

$$\omega(d_1, \ldots, d_k) = \overline{\omega}(x_1, \ldots, x_k) \leftarrow d_1 \leftarrow \cdots \leftarrow d_k \ .$$

Proof. Assume that $\Omega \cap D = \emptyset$ and let $\Omega' = \Omega \cup D$, where the elements of D are treated as nullary symbols. Firstly, we define a mapping $h : T_{\Omega'}(X) \longrightarrow T_{\Omega'}(X)$ for every $v \in D \cup X$ as follows.

$$h(v) = v$$

$$h(\overline{\omega}(t_1, \ldots, t_k)) = \begin{cases} \omega(h(t_1), \ldots, h(t_k)) & \text{, if } h(t_1), \ldots, h(t_k) \in D \\ \overline{\omega}(h(t_1), \ldots, h(t_k)) & \text{, otherwise} \end{cases}$$

Note that $h(t) \in \widehat{T_{\Omega'}}(X_n)$ whenever $t \in \widehat{T_{\Omega'}}(X_n)$. Secondly, let

$$B = D^* \cup \bigcup_{n \in \mathbb{N}_+} D^* \cdot \widehat{T_{\Omega'}}(X_n) \ .$$

Next we define the operation $\leftarrow \, : \ B^2 \longrightarrow B$ for every $w \in D^*$, $b \in B$, $t \in \widehat{T_{\Omega'}}(X_n)$, and $t' \in D \cup \widehat{T_{\Omega'}}(X_n)$ by

$$w \leftarrow b = w \cdot b$$

$$w \cdot t \leftarrow \varepsilon = w \cdot t$$

$$w \cdot t \leftarrow t' \cdot b = w \cdot (h(t \langle\!\langle t' \rangle\!\rangle)) \leftarrow b \ .$$

Roughly speaking, one can understand $\leftarrow$ as function composition where the arguments are lambda-terms and the evaluation (which is done via h) is call-by-value. Next we would like to extend this monoid to a semiring by introducing the addition of the DM-monoid. However, the addition should also be able to sum up terms, hence we first use an abstract addition coming from a semiring for which the DM-monoid is a complete semimodule.

Let $\mathcal{A} = (A, \oplus, \odot)$ be a semiring. We lift the operation $\leftarrow \, : \ B^2 \longrightarrow B$ to an operation $\leftarrow \, : \ A\langle\!\langle B \rangle\!\rangle^2 \longrightarrow A\langle\!\langle B \rangle\!\rangle$ by

$$\psi_1 \leftarrow \psi_2 = \bigoplus_{b_1, b_2 \in B} \big((\psi_1, b_1) \odot (\psi_2, b_2) \big) \, (b_1 \leftarrow b_2) \ .$$

Let the monoid $\mathcal{D} = (D, +)$ be a complete $\mathcal{A}$-semimodule. Then we define the *sum of a series* $\varphi \in A\langle\!\langle D \rangle\!\rangle$ (summed in D) by the mapping $\sum : \ A\langle\!\langle D \rangle\!\rangle \longrightarrow D$ with $\sum \varphi = \sum_{d \in D} (\varphi, d) \cdot d$. For a vector $V \in A\langle\!\langle D \rangle\!\rangle^I$ we let $(\sum V)_i = \sum V_i$. By convenience we identify the series $1_{\mathcal{A}} \, d$ with d.

Proposition 7. *Let the DM-monoid $\mathcal{D} = (D, +, \Omega)$ be a complete $(A, \oplus, \odot)$-semimodule and $\varphi_1, \ldots, \varphi_k \in A\langle\!\langle D \rangle\!\rangle$. Then*

(i) $\sum (\bigoplus_{i \in I} \varphi_i) = \sum_{i \in I} \sum \varphi_i$ *for every family* $(\varphi_i)_{i \in I}$ *of series and*
(ii) $\omega(\sum \varphi_1, \ldots, \sum \varphi_k) = \sum \big(\overline{\omega}(x_1, \ldots, x_k) \leftarrow \varphi_1 \leftarrow \cdots \leftarrow \varphi_k \big)$.

Thus we can construct a semiring with the following properties.

Theorem 8. *For every continuous DM-monoid $\mathcal{D} = (D, +, \Omega)$ there exists a semiring $(C, \oplus, \leftarrow)$ such that $D \cup \Omega X \subseteq C$ and for all $d_1, \ldots, d_k \in D$*

(i) $\omega(d_1, \ldots, d_k) = \overline{\omega}(x_1, \ldots, x_k) \leftarrow d_1 \leftarrow \cdots \leftarrow d_k$,
(ii) $\sum(\bigoplus_{i \in I} d_i) = \sum_{i \in I} d_i$.

Proof. Let $\mathcal{A} = (A, \oplus, \odot)$ be a semiring such that $\mathcal{D}$ is a complete $\mathcal{A}$-semimodule. For example, $\mathcal{A}$ can always be chosen to be $\mathbb{N}_\infty$. By Theorem 6 there exists a monoid $(B, \leftarrow)$ such that Statement (i) holds. Consequently, let $C = A\langle\langle B\rangle\rangle$ and $\leftarrow : C^2 \longrightarrow C$ be the extension of $\leftarrow$ on B. Clearly, $(C, \oplus, \leftarrow)$ is a semiring and by Theorem 6 and Proposition 7 the Statements (i) and (ii) hold.

The semiring $(A\langle\langle B\rangle\rangle, \oplus, \leftarrow)$ constructed in Theorem 8 will be denoted by $G_{\mathcal{A}}(\mathcal{D})$ in the sequel. We note that $G_{\mathcal{A}}(\mathcal{D})$ is complete, because $\mathcal{A}$ is complete (cf. [10]). Hence we are ready to state the first main representation theorem.

Theorem 9. *Let $M_1 = (I_1, \Sigma, \mathcal{D}, F_1, \mu_1)$ be a DM-wta and M_2 be a tst.*

- *There exists a wta $M = (I_1, \Sigma, G_{\mathcal{A}}(\mathcal{D}), F, \mu)$ such that $\|M_1\| = \sum \|M\|$.*
- *There exists a wta M such that $\|M_2\| = \sum \|M\|$.*

Proof. The second statement follows from the first and Proposition 5, so it remains to prove the first statement. Let $F_i = \overline{(F_1)_i}(x_1)$ and

$$\mu_k(\sigma)_{i,(i_1,\ldots,i_k)} = \overline{(\mu_1)_k(\sigma)_{i,(i_1,\ldots,i_k)}}(x_1, \ldots, x_k) \ .$$

Note that again M can be chosen to be deterministic, whenever the input device is deterministic. The main reason for the remaining summation is the fact that we do not know how to define sums like $\overline{\omega}(x_1, \ldots, x_k) + \overline{\omega}'(x_1, \ldots, x_k)$ for $\omega, \omega' \in \Omega_k$. Hence, we finally consider tst, because there we know more about the operations of Ω.

Theorem 10. *Let $\mathcal{A}$ be a completely idempotent semiring and let M_1 be a tst over $\mathcal{A}$. There exists a wta M such that $\|M\| = \|M_1\|$.*

The last theorem admits a trivial corollary.

Corollary 11. *For every bottom-up tree transducer M_1 there exists a wta M such that $\|M\| = \|M_1\|$.*

4 Pumping Lemmata

In this section we would like to demonstrate how to make use of the representation theorem derived in the previous section (Theorem 9). Unfortunately, very few results exist for weighted tree automata over arbitrary semirings (in particular: non-commutative semirings). However, in [13] a pumping lemma for deterministic finite wta is presented and we would like to translate this result to deterministic finite tst and deterministic finite DM-wta.

In this section, let $\mathcal{A} = (A, \oplus, \odot)$ be a semiring and $\mathcal{D} = (D, +, \Omega)$ be a DM-monoid. Let $\mathcal{L}^d_\Sigma(\mathcal{A})$ be the class of deterministically recognizable tree series, i.e., for every $L \in \mathcal{L}^d_\Sigma(\mathcal{A})$ there exists a deterministic finite wta $M = (I, \Sigma, \mathcal{A}, F, \mu)$ such that $L = \|M\|$. Similarly, let $\mathcal{T}^d_{\Sigma,\Delta}(\mathcal{A})$ be the class of deterministically computable t-ts transformations, i.e., for every $\tau \in \mathcal{T}^d_{\Sigma,\Delta}(\mathcal{A})$ there exists a deterministic finite tst $M = (I, \Sigma, \Delta, \mathcal{A}, F, \mu)$ such that $\tau = \|M\|$. Finally, let $\mathcal{L}^d_\Sigma(\mathcal{D})$ be the class of deterministically recognizable DM-monoid tree series, i.e., for every $L \in \mathcal{L}^d_\Sigma(\mathcal{D})$ there exists a deterministic finite DM-wta $M = (I, \Sigma, \mathcal{D}, F, \mu)$ such that $L = \|M\|$.

Firstly, we state the original corollary of [13].

Corollary 12 (Corollary 5.8 of [13]). *Let $L \in \mathcal{L}^d_\Sigma(\mathcal{A})$. There exists $m \in \mathbb{N}$ such that for every tree $t \in \operatorname{supp}(L)$ with $\operatorname{height}(t) \geq m+1$ there exist trees $C, C' \in \widehat{T_\Sigma}(X_1)$ and $t' \in T_\Sigma$, and semiring elements $a, a', b, b', d \in A$ such that*

- $t = C[C'[t']]$,
- $\operatorname{height}(C[t']) \leq m+1$ *and* $C \neq x_1$, *and*
- $(L, C'[C^n[t']]) = a' \odot a^n \odot d \odot b^n \odot b$ *for every* $n \in \mathbb{N}$.

We have already noted that the determinism and finiteness properties are preserved by all our constructions, so given a deterministic finite DM-wta M_1, we can construct a deterministic finite wta M such that $\sum \|M\| = \|M_1\|$ (cf. Theorem 9). Since the addition of the semiring is irrelevant for deterministic devices, we actually obtain $\|M\| = \|M_1\|$. Now we can apply the pumping lemma (Corollary 12) to this wta and thereby obtain a pumping lemma for tree series of $\mathcal{L}^d_\Sigma(\mathcal{D})$.

Theorem 13. *Let $L \in \mathcal{L}^d_\Sigma(\mathcal{D})$ and $\Omega' = \Omega \cup D$. There exists $m \in \mathbb{N}$ such that for every $t \in \operatorname{supp}(L)$ with $\operatorname{height}(t) \geq m+1$ there exist $C, C' \in \widehat{T_\Sigma}(X_1)$, $t' \in T_\Sigma$, and $a, a' \in \widehat{T_{\Omega'}}(X_1)$, and $d \in D$ such that*

- $t = C[C'[t']]$,
- $\operatorname{height}(C[t']) \leq m+1$ *and* $C \neq x_1$, *and*
- $(L, C'[C^n[t']]) = a' \leftarrow a^n \leftarrow d$ *for every* $n \in \mathbb{N}$.

Proof. The statement follows from Corollary 5.8 of [13] and Theorem 9.

With the help of Proposition 5 we can also obtain a pumping lemma for deterministic finite tst in the very same manner.

Theorem 14. *Let $\tau \in \mathcal{T}^d_{\Sigma,\Delta}(\mathcal{A})$ be a t-ts transformation. There exists $m \in \mathbb{N}$ such that for every tree $t \in \operatorname{supp}(T)$ with $\operatorname{height}(t) \geq m+1$ there exist trees $C, C' \in \widehat{T_\Sigma}(X_1)$, $t' \in T_\Sigma$, and $a, a' \in A\langle\langle T_\Delta(X_1)\rangle\rangle$, and $c \in A\langle\langle T_\Delta\rangle\rangle$ such that*

- $t = C[C'[t']]$,
- $\operatorname{height}(C[t']) \leq m+1$ *and* $C \neq x_1$, *and*

– $(\tau, C'[C^n[t']]) = a' \leftarrow a^n \leftarrow c$ *for every* $n \in \mathbb{N}$.

Proof. The statement is an immediate consequence of Proposition 5 and Theorem 13.

Finally, if we instantiate the previous theorem to the Boolean semiring, then we obtain the classical pumping lemma for deterministic bottom-up tree transducers (cf. [16]).

References

1. Rozenberg, G., Salomaa, A., eds.: Handbook of Formal Languages. Volume 1. Springer (1997)
2. Berstel, J.: Transductions and Context-Free Languages. Teubner, Stuttgart (1979)
3. Schützenberger, M.P.: Certain elementary families of automata. In: Proceedings of Symposium on Mathematical Theory of Automata, Polytechnic Institute of Brooklyn (1962) 139–153
4. Eilenberg, S.: Automata, Languages, and Machines – Volume A. Volume 59 of Pure and Applied Mathematics. Academic Press (1974)
5. Kuich, W., Salomaa, A.: Semirings, Automata, Languages. Volume 5 of EATCS Monographs on Theoretical Computer Science. Springer (1986)
6. Comon, H., Dauchet, M., Gilleron, R., Jacquemard, F., Lugiez, D., Tison, S., Tommasi, M.: Tree automata – techniques and applications. Available on: `http://www.grappa.univ-lille3.fr/tata` (1997)
7. Rounds, W.C.: Mappings and grammars on trees. Mathematical Systems Theory **4** (1970) 257–287
8. Thatcher, J.W.: Generalized^2 sequential machine maps. Journal of Computer and System Sciences **4** (1970) 339–367
9. Bozapalidis, S.: Equational elements in additive algebras. Theory of Computing Systems **32** (1999) 1–33
10. Kuich, W.: Formal power series over trees. In Bozapalidis, S., ed.: Proceedings of 3rd DLT 1997, Aristotle University of Thessaloniki (1997) 61–101
11. Engelfriet, J., Fülöp, Z., Vogler, H.: Bottom-up and top-down tree series transformations. Journal of Automata, Languages and Combinatorics **7** (2002) 11–70
12. Fülöp, Z., Vogler, H.: Tree series transformations that respect copying. Theory of Computing Systems **36** (2003) 247–293
13. Borchardt, B.: A pumping lemma and decidability problems for recognizable tree series. Acta Cybernetica (2004) to appear.
14. Kuich, W.: Tree transducers and formal tree series. Acta Cybernetica **14** (1999) 135–149
15. Thatcher, J.W., Wagner, E.G., Wright, J.B.: Initial algebra semantics and continuous algebra. Journal of the ACM **24** (1977) 68–95
16. Engelfriet, J.: Some open questions and recent results on tree transducers and tree languages. In Book, R.V., ed.: Formal Language Theory – Perspectives and Open Problems. Academic Press (1980) 241–286