Tree-Series-to-Tree-Series Transformations

Andreas Maletti*

International Computer Science Institute
1947 Center Street, Suite 600, Berkeley CA-94704, USA
maletti@icsi.berkeley.edu

Abstract. We investigate the tree-series-to-tree-series (ts-ts) transformation computed by tree series transducers. Unless the used semiring is complete, this transformation is, in general, not well-defined. In practice, many used semirings are not complete (like the probability semiring). We establish a syntactical condition that guarantees well-definedness of the ts-ts transformation in arbitrary commutative semirings. For positive (*i. e.*, zero-sum and zero-divisor free) semirings the condition actually characterizes the well-definedness, so that well-definedness is decidable in this scenario.

1 Introduction

Tree series transducers [1,2] are a generalization of tree transducers [3,4,5,6,7]. The framework Tiburon [8] implements a generalization of top-down tree series transducers [2] using various weight structures such as the Boolean semiring $(\{0,1\}, \vee, \wedge)$ and the probability semiring $(\mathbb{R}, +, \cdot)$. Such tree series transducers compute both a tree-to-tree-series (t-ts) and a tree-series-to-tree-series (ts-ts) transformation, where a tree series is a mapping assigning a weight to each tree. The t-ts transformation is always well-defined, but the ts-ts transformation is well-defined only for complete semirings [9,10] such as the Boolean semiring. However, for the probability semiring the ts-ts transformation need not be well-defined because infinite sums might occur. Of course, some incomplete semirings (*e. g.*, positive semirings) can be extended by a new element ∞, which is the result of all nontrivial infinite sums. However, such a definition is clearly not practical and does not work for the probability semiring.

A standard application of the ts-ts transformation is the computation of the image of a recognizable tree series [11,12,13,14]. This is, for example, used to translate a language model (parses of an input sentence) to a language model (resp., parses of output sentences) in another language. For some tree series transducers the image is again a recognizable tree series [15,16]. In fact, the image operation is implemented in Tiburon for the Boolean semiring. However, in the probability semiring, the image operation is only meaningful if the ts-ts transformation is well-defined.

* This work was supported by a fellowship within the Postdoc-Programme of the German Academic Exchange Service (DAAD).

O.H. Ibarra and B. Ravikumar (Eds.): CIAA 2008, LNCS 5148, pp. 132–140, 2008.

In this contribution we investigate for which tree series transducers the ts-ts transformation is well-defined following the approach of [17,18] for weighted finite-state transducers. To this end, we develop a general notion of convergence that can serve as a baseline for all semirings. More refined notions for particular semirings can be derived in the same manner. Thereafter we present a syntactical condition, which in general, guarantees that the ts-ts transformation is well-defined (using the baseline notion of convergence mentioned). In fact, the condition is such that we obtain a characterization for certain tree series transducers over positive (*i. e.*, zero-sum and zero-divisor free) semirings. This yields that well-definedness of the ts-ts transformation is decidable for certain tree series transducers over positive semirings. This also applies to tree series transducers over the BOOLEAN semiring (*i. e.*, tree transducers).

2 Preliminaries

The nonnegative integers are denoted by $\mathbb{N}$ and $\mathbb{N}_+ = \mathbb{N} \setminus \{0\}$. We use $[k, n]$ for $\{i \mid k \leqslant i \leqslant n\}$ where the i are either integers or reals depending on the context. In the former case, we abbreviate $[1, n]$ to $[n]$. An *alphabet* is a finite set of *symbols*. A *ranked alphabet* is an alphabet Σ together with a mapping $\mathrm{rk}\colon \Sigma \to \mathbb{N}$, which assigns to each symbol a *rank*. The set of symbols of rank k is denoted by Σ_k. For convenience we assume fixed sets $\mathrm{X} = \{\mathrm{x}_i \mid i \in \mathbb{N}_+\}$ and $\mathrm{Z} = \{\mathrm{z}_i \mid i \in \mathbb{N}_+\}$ of *variables*. For $k \in \mathbb{N}$ we use $\mathrm{X}_k = \{\mathrm{x}_i \mid i \in [k]\}$ and $\mathrm{Z}_k = \{\mathrm{z}_i \mid i \in [k]\}$. Given $V \subseteq \mathrm{X} \cup \mathrm{Z}$, the set $T_\Sigma(V)$ of *Σ-trees indexed by V* is the smallest set T such that $V \subseteq T$ and for every $\sigma \in \Sigma_k$ and $t_1, \ldots, t_k \in T$ also $\sigma(t_1, \ldots, t_k) \in T$. We generally assume that $\mathrm{X} \cup \mathrm{Z}$ is disjoint with any considered ranked alphabet, so we usually write α instead of $\alpha()$ whenever $\alpha \in \Sigma_0$. Moreover, we also use T_Σ for $T_\Sigma(\emptyset)$. Let $t, t_1, \ldots, t_k \in T_\Sigma(\mathrm{Z})$. We denote by $t[t_1, \ldots, t_k]$ the tree obtained from t by replacing for every $i \in [k]$ every z_i-leaf in t by the tree t_i. The tree t is *nondeleting* (resp., *linear*) in $V \subseteq \mathrm{Z}$, if each $v \in V$ occurs at least (resp., at most) once in t. The set of variables occurring in t is $\mathrm{var}(t)$ and the *size* of t (*i. e.*, the number of nodes in t) is $\mathrm{size}(t)$. Finally, the *height* of a tree is inductively defined by $\mathrm{height}(v) = 1$ for every $v \in V$ and $\mathrm{height}(\sigma(t_1, \ldots, t_k)) = 1 + \max\{\mathrm{height}(t_i) \mid i \in [k]\}$ for every $\sigma \in \Sigma_k$ and $t_1, \ldots, t_k \in T_\Sigma(V)$.

An algebraic structure $(A, +)$ is a *monoid* if $+$ is an associative (binary) operation on A that permits a neutral element. A *(commutative) semiring* $(A, +, \cdot)$ consists of two commutative monoids $(A, +)$ and $(A, \cdot)$ such that $\cdot$ distributes over $+$ and the neutral element 0 of $(A, +)$ is absorbing with respect to $\cdot$ (*i. e.*, $a \cdot 0 = 0 = 0 \cdot a$ for every $a \in A$). The neutral element of an additive operation is usually denoted by 0 and that of multiplicative operation by 1. We also use the summation $\sum_{i \in I} a_i$ for an index set I and a family $(a_i \mid i \in I)$ of semiring elements. Such a summation is *well-defined* if $a_i = 0$ for almost all $i \in I$. The actual sum is then defined in the obvious way. A semiring $\mathcal{A} = (A, +, \cdot)$ is *zero-sum free*, whenever $a + b = 0$ implies that $a = 0$ for every $a, b \in A$, and *zero-divisor* *free*, whenever $a \cdot b = 0$ implies that $0 \in \{a, b\}$. A zero-sum and zero-divisor free semiring is *positive*.

Let $\mathcal{A} = (A, +, \cdot)$ be a semiring. Every mapping $\varphi\colon T \to A$ for some $T \subseteq T_\Sigma(V)$ is a *tree series*. We denote the set of those by $\mathcal{A}\langle\langle T \rangle\rangle$. We usually write the *coefficient* $\varphi(t)$ *of* t *in* φ as (φ, t). Moreover, we write φ as the formal sum $\sum_{t \in T} (\varphi, t)\, t$. We extend both operations of $\mathcal{A}$ componentwise to tree series, *i. e.*, $(\varphi + \psi, t) = (\varphi, t) + (\psi, t)$ for every $\varphi, \psi \in \mathcal{A}\langle\langle T \rangle\rangle$ and $t \in T$. The *support of* φ is $\mathrm{supp}(\varphi) = \{t \mid (\varphi, t) \neq 0\}$. The set of tree series with finite support is denoted by $\mathcal{A}\langle T \rangle$. For every $a \in A$, the tree series $\widetilde{a}$ is such that $(\widetilde{a}, t) = a$ for every $t \in T$. The tree series φ is *nondeleting* (resp., *linear*) in V, if every $t \in \mathrm{supp}(\varphi)$ is nondeleting (resp., linear) in V. We use $\mathrm{var}(\varphi)$ as a shorthand for $\bigcup_{t \in \mathrm{supp}(\varphi)} \mathrm{var}(t)$.

Let $\varphi \in \mathcal{A}\langle T_\Delta(\mathrm{Z}) \rangle$ and $\psi_1, \dots, \psi_k \in \mathcal{A}\langle T_\Delta(\mathrm{Z}) \rangle$. The *pure substitution* [19,2] of $(\psi_1, \dots, \psi_k)$ into φ is defined by

$$\varphi \leftarrow (\psi_1, \dots, \psi_k) = \sum_{t, t_1, \dots, t_k \in T_\Delta(Z)} (\varphi, t)(\psi_1, t_1) \cdots (\psi_k, t_k)\, t[t_1, \dots, t_k] \enspace .$$

Let $\mathcal{A}$ be a semiring, Σ and Δ be ranked alphabets, and Q be a finite set. A *(polynomial) representation* [2] is a family $\mu = (\mu_k \mid k \in \mathbb{N})$ of mappings $\mu_k\colon \Sigma_k \to \mathcal{A}\langle T_\Delta(\mathrm{Z}) \rangle^{Q \times (Q \times \mathrm{X}_k)^*}$ such that for every $\sigma \in \Sigma_k$ and $q \in Q$

(i) $\mu_k(\sigma)_{q,w} \in \mathcal{A}\langle T_\Delta(\mathrm{Z}_{|w|}) \rangle$ for every $w \in (Q \times \mathrm{X}_k)^*$ and
(ii) $\mu_k(\sigma)_{q,w} = \widetilde{0}$ for almost all $w \in (Q \times \mathrm{X}_k)^*$.

A *(polynomial) tree series transducer* [1,2] is a tuple $(Q, \Sigma, \Delta, \mathcal{A}, I, \mu)$ such that μ is a representation and $I \subseteq Q$. It is *top-down* (resp., *bottom-up*) [2] if $\mu_k(\sigma)_{q,w}$ is nondeleting and linear in $\mathrm{Z}_{|w|}$ [resp., if there exist $q_1, \dots, q_k \in Q$ such that $w = (q_1, \mathrm{x}_1) \cdots (q_k, \mathrm{x}_k)$] for every $\sigma \in \Sigma_k$, $q \in Q$, and $w \in (Q \times X_k)^*$ such that $\mu_k(\sigma)_{q,w} \neq \widetilde{0}$. Let $h_\mu\colon T_\Sigma \to \mathcal{A}\langle\langle T_\Delta \rangle\rangle^Q$ be defined for every $\sigma \in \Sigma_k$, $t_1, \dots, t_k \in T_\Sigma$, and $q \in Q$ by

$$h_\mu\big(\sigma(t_1, \dots, t_k)\big)_q = \sum_{\substack{w \in (Q \times \mathrm{X}_k)^*, \\ w = (q_1, \mathrm{x}_{i_1}) \cdots (q_n, \mathrm{x}_{i_n})}} \mu_k(\sigma)_{q,w} \leftarrow \big(h_\mu(t_{i_1})_{q_1}, \dots, h_\mu(t_{i_n})_{q_n}\big) \enspace .$$

The transducer M computes the *tree-to-tree-series transformation* (t-ts transformation) $\tau_M\colon T_\Sigma \to \mathcal{A}\langle\langle T_\Delta \rangle\rangle$ defined by $\tau_M(t) = \sum_{q \in I} h_\mu(t)_q$ for every $t \in T_\Sigma$. Both h_μ and the t-ts transformation τ_M are well-defined. Finally, the *tree-series-to-tree-series transformation* (ts-ts transformation) computed by M is $\tau_M(\varphi) = \sum_{t \in T_\Sigma} (\varphi, t) \cdot \tau_M(t)$ for every $\varphi \in \mathcal{A}\langle\langle T_\Sigma \rangle\rangle$, whenever this sum is well-defined. We say that τ_M is well-defined whenever $\tau_M(\varphi)$ is well-defined for every $\varphi \in \mathcal{A}\langle\langle T_\Sigma \rangle\rangle$.

3 Convergence

In this section, we will explore when the ts-ts transformation of a tree series transducer $M = (Q, \Sigma, \Delta, \mathcal{A}, I, \mu)$ is well-defined. Roughly speaking, it is well-defined if every output tree $u \in T_\Delta$ can be generated [*i. e.*, $u \in \mathrm{supp}(\tau_M(t))$] by only finitely many input trees $t \in T_\Sigma$. Note that our definition of well-definedness works in any semiring; for particular semirings like $(\mathbb{R}, +, \cdot, 0, 1)$ other notions of well-definedness (or equivalently, convergence) might be more realistic. However, those more refined notions typically include our notion of well-definedness (*i. e.*, any sum that is well-defined according to our definition is also well-defined in the refined setting and the sums coincide), so that our approach can be seen as a general baseline. We first show that τ_M is well-defined if and only if $\tau_M(\widetilde{1})$ is well-defined. Thus, subsequent investigations need not consider the actual input tree series.

Proposition 1. *The ts-ts transformation τ_M is well-defined if and only if $\tau_M(\widetilde{1})$ is well-defined.*

Proof. Let $\varphi \in \mathcal{A}\langle\langle T_\Sigma \rangle\rangle$ and $u \in T_\Delta$. One direction is trivial. In the other direction, the sum $\tau_M(\widetilde{1})$ is well-defined by assumption. Hence, $(\tau_M(t), u) = 0$ for almost all $t \in T_\Sigma$. Thus, $\tau_M(\varphi)$ is well-defined. □

Let us take a closer look at $\tau_M(\widetilde{1})$. By definition, it is $\sum_{t \in T_\Sigma} \tau_M(t)$. This is well-defined if it is not possible to transform large (with respect to the size) input trees to small output trees. Let us introduce the notion of convergence [18] that we will use. For every $\varphi \in \mathcal{A}\langle\langle T_\Delta(\mathrm{Z}) \rangle\rangle$ let $\|\varphi\| = \max_{t \in \mathrm{supp}(\varphi)} \mathrm{size}(t)^{-1}$. We call $\|\varphi\|$ the *norm of* φ. Intuitively, the norm of φ is the inverse of the size of a smallest tree in the support of φ. Thus, the norm of $\widetilde{0}$ is 0.

Proposition 2. *For every $\varphi, \psi \in \mathcal{A}\langle\langle T_\Delta(\mathrm{Z}) \rangle\rangle$*

(i) $\|\varphi\| = 0$ *if and only if* $\varphi = \widetilde{0}$.
(ii) $\|\varphi + \psi\| \leqslant \|\varphi\| + \|\psi\|$.

Actually, it can be shown that $\|\cdot\|$ is a monoid-homomorphism from the monoid $(\mathcal{A}\langle\langle T_\Delta(\mathrm{Z}) \rangle\rangle, +)$ to $([0,1], \max)$ if $\mathcal{A}$ is zero-sum free. We derive the distance $d_{\|\cdot\|}$ on $\mathcal{A}\langle\langle T_\Delta(\mathrm{Z}) \rangle\rangle$, which is given by $d_{\|\cdot\|}(\varphi, \psi) = \big|\, \|\varphi\| - \|\psi\| \,\big|$ for every $\varphi, \psi \in \mathcal{A}\langle\langle T_\Delta(\mathrm{Z}) \rangle\rangle$.

Proposition 3. *The distance $d_{\|\cdot\|}$ defines a pseudometric on $\mathcal{A}\langle\langle T_\Delta(\mathrm{Z}) \rangle\rangle$.*

With the help of this pseudometric, we can now introduce the usual notion of Cauchy-convergence for sequences of tree series.

Definition 4. *Let $\Psi = (\psi_i \mid i \in \mathbb{N})$ be a family of $\psi_i \in \mathcal{A}\langle\langle T_\Delta(\mathrm{Z}) \rangle\rangle$. It* converges (using the pseudometric $d_{\|\cdot\|}$) *if*

$$(\exists \psi \in \mathcal{A}\langle\langle T_\Delta(\mathrm{Z}) \rangle\rangle)(\forall \epsilon > 0)(\exists j_\epsilon \in \mathbb{N})(\forall j \geqslant j_\epsilon) \colon d_{\|\cdot\|}(\psi_j, \psi) < \epsilon \ .$$

If Ψ converges, then ψ in the above display is a limit of Ψ *and we say that Ψ converges to ψ or symbolically $\Psi \to \psi$.*

Convergence to $\widetilde{0}$ will play a central role. In fact, Ψ converges to $\widetilde{0}$ if

$$(\forall n \in \mathbb{N})(\exists j_n \in \mathbb{N})(\forall j \geqslant j_n)\colon \min_{t \in \operatorname{supp}(\psi_j)} \operatorname{size}(t) > n \enspace .$$

Let $T = (t_i \mid i \in \mathbb{N})$ be a family of $t_i \in T_\Sigma$. It is an *enumeration* of T_Σ if for every $t \in T_\Sigma$ there exists exactly one $i \in \mathbb{N}$ such that $t_i = t$, and it is *size-compliant* if $\operatorname{size}(t_i) \leqslant \operatorname{size}(t_j)$ for all $i \leqslant j$. We write $\tau_M(T)$ for the family $(\tau_M(t_i) \mid i \in \mathbb{N})$. Next we characterize when $\tau_M(\widetilde{1})$ is well-defined in terms of size-compliant enumerations.

Theorem 5. *The following are equivalent:*

(i) τ_M *is well-defined.*
(ii) $\tau_M(T) \to \widetilde{0}$ *for every size-compliant enumeration* T *of* T_Σ*.*
(iii) $\tau_M(T) \to \widetilde{0}$ *for some size-compliant enumeration* T *of* T_Σ*.*

Proof. The existence of at least one size-compliant enumeration of T_Σ is self-evident, so (ii) clearly implies (iii). Let us assume that there exists a size-compliant enumeration $T = (t_i \mid i \in \mathbb{N})$ such that $\tau_M(T)$ converges to $\widetilde{0}$. We know that for every $n \in \mathbb{N}$ there exists a $j_n \in \mathbb{N}$ such that for all $j \geqslant j_n$ we have that $\min_{u \in \operatorname{supp}(\tau_M(t_j))} \operatorname{size}(u) > n$, or equivalently, $u \notin \operatorname{supp}(\tau_M(t_j))$ for all $u \in T_\Delta$ with $\operatorname{size}(u) \leqslant n$. In particular, for every $u \in T_\Delta$ there exists $n_u \in \mathbb{N}$ such that $u \notin \operatorname{supp}(\tau_M(t_n))$ for all $n \geqslant n_u$. Thus, $\tau_M(\widetilde{1})$ and by Proposition 1 also τ_M are well-defined.

Conversely, suppose that τ_M and hence $\tau_M(\widetilde{1})$ are well-defined (see Proposition 1). There exists a finite subset $S_u \subseteq T_\Sigma$ for every tree $u \in T_\Delta$ such that $u \notin \operatorname{supp}(\tau_M(t))$ for every $t \notin S_u$. Let $n \in \mathbb{N}$ and $T = (t_i \mid i \in \mathbb{N})$ be a size-compliant enumeration of T_Σ. Let $U_n = \{u \in T_\Delta \mid \operatorname{size}(u) \leqslant n\}$ and $S_n = \bigcup_{u \in U_n} S_u$. Clearly, U_n and thus also S_n are finite. Finally, we let $m_n = \max_{t \in S_n} \operatorname{size}(t) + 1$ and j_n be an index such that $\operatorname{size}(t_{j_n}) \geqslant m_n$. It remains to prove that $\min_{u \in \operatorname{supp}(\tau_M(t_j))} \operatorname{size}(u) > n$ for every $j \geqslant j_n$. Suppose that $u \in \operatorname{supp}(\tau_M(t_j))$ and $\operatorname{size}(u) \leqslant n$. Thus $u \in U_n$. By this, we obtain that $t_j \in S_u$ and $t_j \in S_n$. It follows that $m_n \geqslant \operatorname{size}(t_j) + 1$. By the size-compliance condition, $\operatorname{size}(t_j) \geqslant \operatorname{size}(t_{j_n}) \geqslant m_n$. With the previous inequality, we obtain $\operatorname{size}(t_j) \geqslant \operatorname{size}(t_j) + 1$. Thus, there exists no $u \in \operatorname{supp}(\tau_M(t_j))$ with $\operatorname{size}(u) \leqslant n$, which proves that $\tau_M(T) \to \widetilde{0}$. ☐

The previous theorem is clear if $\mathcal{A}$ is zero-sum free, but in other cases one might be tempted to assume that the theorem only holds because of our peculiar (or even deficient) definition of well-defined sums. Let us show on an example that this is indeed not the case. Let $\Sigma = \Delta = \{\gamma^{(1)}, \alpha^{(0)}\}$ and $\mathcal{A} = \mathbb{Z}$. Moreover, let $\tau_M(t) = (-1)^{|t|_\gamma}\, \alpha$. Now one might argue that $\tau_M(\widetilde{1})$ is well-defined and equal to $\widetilde{0}$ because $\tau_M(\gamma^n(\alpha)) + \tau_M(\gamma^{n+1}(\alpha)) = \widetilde{0}$ for every even n. However, the last property also holds for each odd n, which yields $\tau_M(\widetilde{1}) = \tau_M(\alpha) + \sum_{t \in T_\Sigma \setminus \{\alpha\}} \tau_M(t) = \tau_M(\alpha)$. Thus, we argued for two different results of the sum, which shows that it is not well-defined.

4 Towards a Syntactical Property

Next, we present a syntactic condition that guarantees that the ts-ts transformation computed by a tree series transducer is well-defined. To this end, let $M = (Q, \Sigma, \Delta, \mathcal{A}, I, \mu)$ be a tree series transducer. Note that we could reduce the problem to unweighted tree transducers, but we avoid this for two reasons: (i) It is rather unintuitive that $\bigvee_{i \in \mathbb{N}} 1$ is not well-defined in the Boolean semiring $(\{0,1\}, \vee, \wedge)$ and (ii) we lack the space to introduce them (using the standard set notation). We generally follow the approach of [17,18] by the analysis is slightly more complicated by the tree structure. First we introduce some important notions like the dependency relations $P, R \subseteq Q \times Q$. For every $p, q \in Q$, let $(p, q) \in P$ (resp., $(p, q) \in R$) if $\mathrm{z}_i \in \mathrm{supp}(\mu_k(\sigma)_{p,w}$ (resp., $\mathrm{supp}(\mu_k(\sigma)_{p,w}) \neq \emptyset$) for some $\sigma \in \Sigma_k$ and $w \in (Q \times \mathrm{X}_k)^*$ such that $w_j = (q, \mathrm{x}_i)$ for some $1 \leqslant j \leqslant |w|$. Let $\sqsubset$ and $\sqsubseteq$ (resp., $\prec$ and $\preceq$) be the transitive and reflexive, transitive closure of P (resp., of R), respectively. Note that in general $\sqsubseteq$ and $\preceq$ are not partial orders. Then the following definitions are natural (note that our reading is top-down).

Definition 6. *Let $q \in Q$.*

- *If $q \sqsubset q$ (resp., $q \prec q$), then q is* circular *(resp.,* self-replicating*).*
- *If there exists $p \in I$ such that $p \preceq q$, then q is* accessible.
- *If there exist $p \in Q$ and $\alpha \in \Sigma_0$ such that $\mu_0(\alpha)_{p,\varepsilon} \neq \widetilde{0}$ and $q \preceq p$, then q is* co-accessible.

The tree series transducer M is reduced *if every state is accessible and co-accessible. Finally, M is* non-circular *if no state $q \in Q$ is circular.*

Note that τ_M is trivially well-defined if M has no self-replicating state (the latter can easily be checked). In the sequel, we assume that M has at least one self-replicating state. It is also obvious that we can construct a reduced tree series transducer M' that is equivalent to M. We simply remove all states that are not accessible or not co-accessible. It should be clear that this procedure does not change the computed tree series.

Proposition 7. *There exists a reduced tree series transducer M' with $\tau_M = \tau_{M'}$.*

Next, we introduce an essential notion: *deletion points*. A deletion point is a pair (p, q) of states such that one of the transitions into p deletes a subtree potentially processed in q.

Definition 8. *We say that $(p, q) \in Q^2$ is a* deletion point *if there exist $\sigma \in \Sigma_k$, $w \in (Q \times \mathrm{X}_k)^*$, $u \in \mathrm{supp}(\mu_k(\sigma)_{p,w})$, and $i \in [k]$ such that*

- *there does not exist $1 \leqslant j \leqslant |w|$ and $r \in Q$ such that $w_j = (r, \mathrm{x}_i)$, or*
- *$\mathrm{z}_j \notin \mathrm{var}(u)$ for some $1 \leqslant j \leqslant |w|$ such that $w_j = (q, \mathrm{x}_i)$.*

The conditions could be called input- *and* output-deleting*, respectively.*

Note that top-down and bottom-up tree series transducers have a deletion point if and only if they are deleting [2]. Note that if a top-down tree transducer has the deletion point (p, q), then it also has the deletion point (p, r) for every $r \in Q$. Let us illustrate the notion on a small example.

Example 9. Let $M = (\{\star, \bot\}, \Sigma, \Sigma, \mathbb{N}, \{\star\}, \mu)$ be the tree series transducer with $\Sigma = \{\sigma^{(2)}, \alpha^{(0)}\}$ and

$$\mu_0(\alpha)_{p,\varepsilon} = 1\,\alpha \qquad \mu_2(\sigma)_{\bot,(\bot,\mathrm{x}_1)(\bot,\mathrm{x}_2)} = 1\,\sigma(\mathrm{z}_1, \mathrm{z}_2)$$
$$\mu_2(\sigma)_{\star,(\star,\mathrm{x}_1)(\bot,\mathrm{x}_2)} = 1\,\sigma(\mathrm{z}_1, \alpha) \qquad \mu_2(\sigma)_{\star,(\bot,\mathrm{x}_1)(\star,\mathrm{x}_2)} = 1\,\sigma(\alpha, \mathrm{z}_2)$$

for every $p, q \in \{\star, \bot\}$. Then only $(\star, \bot)$ is a deletion point.

Definition 10 (see, *e. g.*, [18]). *The tree series transducer M is* regulated *if it is non-circular and there exists no deletion point (p, q) such that $q \preceq r$ for some self-replicating $r \in Q$.*

Note that it is clearly decidable whether a tree series transducer is regulated. A regulated top-down tree series transducer is nondeleting [2]. This is due to the fact that a deleting top-down tree series transducer has a deletion point (p, q) and thus also the deletion point (p, r) where r is a self-replicating state.

Theorem 11. *Let M be a regulated tree series transducer. Then τ_M is well-defined.*

Proof. Let $M = (Q, \Sigma, \Delta, \mathcal{A}, I, \mu)$. By Theorem 5, it is sufficient to show that for an arbitrary size-compliant enumeration $T = (t_i \mid i \in \mathbb{N})$ the family $\tau_M(T)$ converges to $\widetilde{0}$. Let $\mathrm{mx} = \max\{k \mid \Sigma_k \neq \emptyset\}$ and $n = \mathrm{card}(Q)$. We will prove that $\lfloor \mathrm{height}(t)/n \rfloor - n \leqslant \mathrm{height}(u)$ for every $t \in T_\Sigma$ and $u \in \mathrm{supp}(\tau_M(t))$. Consider a maximal path in t (which defines the height). Since M is non-circular, it may erase at most $n - 1$ input symbols along this path before it produces output. It might also decide to delete the translation incurred along a suffix of the path. However, the length of such a suffix is limited by n because otherwise M has a deletion point that leads to a self-replicating state. Note that if M is a top-down tree series transducer, then it may not delete (because regulated implies nondeletion). Thus, in this case the bound could be improved to $\lfloor \mathrm{height}(t)/n \rfloor \leqslant \mathrm{height}(u)$. The formal proof of both bounds is straightforward and hence omitted. With the given lower bound, it is clear that $\tau_M(T)$ converges to $\widetilde{0}$ because $\mathrm{height}(u) \leqslant \mathrm{size}(u)$ for every $u \in T_\Delta$ and $\mathrm{size}(t) \leqslant \mathrm{mx}^{\mathrm{height}(t)}$ for every $t \in T_\Sigma$. Thus, τ_M is well-defined. □

We will show the converse only for positive semirings. The main benefit of this approach is that the problem can essentially be reduced to unweighted transducers. We need an additional notion. The tree series transducer M is *input-linear* if for every $q \in Q$, $\sigma \in \Sigma_k$, and $w \in (Q \times \mathrm{X}_k)^*$ such that $\mu_k(\sigma)_{q,w} \neq \widetilde{0}$ there exists at most one $1 \leqslant j \leqslant |w|$ such that $w_j = (p, x)$ for every $x \in \mathrm{X}_k$. Note that bottom-up implies input-linear. The following lemma shows that every tree series transducer can be turned into an input-nondeleting one (see Definition 8). In fact, we will only need it for input-linear tree series transducers.

Lemma 12 (see [20, Lemma 1(1)]). *If M is input-linear, then there exists a bottom-up tree series transducer M' such that $\tau_{M'} = \tau_M$.*

Proof. It follows directly by reconsidering the proof of [20, Lemma 1(1)]. The top-down tree series transducer constructed in this proof will be the identity if M is input-linear (as already noted before [20, Theorem 4]). Finally, note that the completeness-assumption is not necessary in our case because our tree series transducers are always polynomial [20]. □

Consequently, we will only deal with bottom-up tree series transducers. For those there exists a decomposition result [2, Lemma 5.6], which states that every bottom-up tree series transducer can be decomposed into a relabeling tree series transducer and a $\{0, 1\}$-weighted homomorphism tree series transducer (see [2] for the definitions of those notions). Roughly speaking, the relabeling tree series transducer annotates each node of the input tree by an applicable entry of μ. Such relabeled input trees are called runs. The homomorphism then simply evaluates the run thereby creating the output tree. We use this decomposition in the following informal argument.

Lemma 13. *Let M be a reduced bottom-up tree series transducer over a positive semiring. If τ_M is well-defined, then M is regulated.*

Proof. Suppose that $M = (Q, \Sigma, \Delta, \mathcal{A}, I, \mu)$ is not regulated. Since $\mathcal{A}$ is positive, we restrict ourselves to the unweighted (*i. e.*, Boolean-semiring weighted) bottom-up tree transducer M' obtained by replacing every nonzero semiring coefficient in μ by 1. By a minor extension of [21, Corollary 3] we have $\mathrm{supp}(\tau_{M'}(t)) = \mathrm{supp}(\tau_M(t))$ for every $t \in T_\Sigma$. We will identify M and M' in the following discussion. If M has a deletion point (p, q), then there exists a subtree u of a run, which is deleted by the evaluation homomorphism, because p is accessible and co-accessible. Note that we can replace u by any run that arrives in the state p at the root. If there exists a self-replicating state r such that $p \preceq r$, then it is immediately clear that there exist infinitely many such runs, and consequently, infinitely many suitable input trees. Since the subrun is deleted all those input trees can be transformed to the same output tree. On the other hand, if M is circular, then we can transform infinitely many input trees into the same output tree by using the circle any number of times. The formal proof is again straightforward and omitted. □

Theorem 14. *Let M be a reduced input-linear tree series transducer over a positive semiring. Then τ_M is well-defined if and only if M is regulated.*

Proof. It follows from Theorem 11 and Lemmata 12 and 13. □

References

1. Kuich, W.: Tree transducers and formal tree series. Acta Cybernet. **14**(1) (1999) 135–149

2. Engelfriet, J., Fülöp, Z., Vogler, H.: Bottom-up and top-down tree series transformations. J. Autom. Lang. Combin. **7**(1) (2002) 11–70
3. Rounds, W.C.: Mappings and grammars on trees. Math. Systems Theory **4**(3) (1970) 257–287
4. Thatcher, J.W.: Generalized^2 sequential machine maps. J. Comput. System Sci. **4**(4) (1970) 339–367
5. Thatcher, J.W.: Tree automata—an informal survey. In: Currents in the Theory of Computing. Prentice Hall (1973) 143–172
6. Engelfriet, J.: Bottom-up and top-down tree transformations—a comparison. Math. Systems Theory **9**(3) (1975) 198–231
7. Engelfriet, J.: Top-down tree transducers with regular look-ahead. Math. Systems Theory **10**(1) (1977) 289–303
8. May, J., Knight, K.: Tiburon: A weighted tree automata toolkit. In: Proc. 11th Int. Conf. Implementation and Application of Automata. Volume 4094 of LNCS., Springer (2006) 102–113
9. Hebisch, U., Weinert, H.J.: Semirings—Algebraic Theory and Applications in Computer Science. World Scientific (1998)
10. Golan, J.S.: Semirings and their Applications. Kluwer Academic, Dordrecht (1999)
11. Bozapalidis, S., Louscou-Bozapalidou, O.: The rank of a formal tree power series. Theoret. Comput. Sci. **27**(1–2) (1983) 211–215
12. Bozapalidis, S.: Equational elements in additive algebras. Theory Comput. Systems **32**(1) (1999) 1–33
13. Kuich, W.: Formal power series over trees. In: Proc. 3rd Int. Conf. Developments in Language Theory, Aristotle University of Thessaloniki (1998) 61–101
14. Borchardt, B., Vogler, H.: Determinization of finite state weighted tree automata. J. Autom. Lang. Combin. **8**(3) (2003) 417–463
15. Kuich, W.: Full abstract families of tree series I. In: Jewels Are Forever. Springer (1999) 145–156
16. Maletti, A.: Pure and o-substitution. Int. J. Found. Comput. Sci. **18**(4) (2007) 829–845
17. Salomaa, A., Soittala, M.: Theoretic Aspects of Formal Power Series. Springer (1978)
18. Kuich, W., Salomaa, A.: Semirings, Automata, Languages. Volume 5 of Monographs in Theoretical Computer Science. An EATCS Series. Springer (1986)
19. Bozapalidis, S.: Context-free series on trees. Inform. and Comput. **169**(2) (2001) 186–229
20. Maletti, A.: The power of tree series transducers of type I and II. In: Proc. 9th Int. Conf. Developments in Language Theory. Volume 3572 of LNCS., Springer (2005) 338–349
21. Maletti, A.: Hierarchies of tree series transformations revisited. In: Proc. 10th Int. Conf. Developments in Language Theory. Volume 4036 of LNCS., Springer (2006) 215–225